KARNATAKA LAW SOCIETY'S

GOGTE INSTITUTE OF TECHNOLOGY

UDYAMBAG BELAGAVI -590008

KARNATAKA, INDIA.

A Course Project Report on

DATABASE FOR A PHARMACY

Submitted for the requirements of 4th semester B.E. in CSE

for "**DATABASE MANAGEMENT SYSTEM (18CS43)**"

Submitted by

NAME	USN
SHAMANT MYAGERI	2GI20CS134
SHRADHA MALLIKARJUN PATIL	2GI20CS144
SRUSHTI B MUDENNAVAR	2GI20CS158
YASH HEREKAR	2GI20CS184

Under the guidance of

Prof. Padma Dandannavar

Asst. Prof., Dept. of CSE

Academic Year 2021 - 2022 (Even semester)

KARNATAKA LAW SOCIETY'S

GOGTE INSTITUTE OF TECHNOLOGY

Udyambag Belagavi -590008

Karnataka, India.

Department of computer science and engineering

Certificate

This is to certify that the Course Project work titled **"Database for a Pharmacy"** carried out by **Students Shamant Myageri (2GI20CS134), Shradha Mallikarjun Patil (2GI20CS144), Srushti B Mudennavar (2GI20CS158)** and **Yash Herekar (2GI20CS184)** for **Subject (18CS43) Theory Course** is submitted in partial fulfilment of the requirements for 4th semester B.E. in **COMPUTER SCIENCE AND ENGINEERING,** Visvesvaraya Technological University, Belagavi. It is certified that all corrections/ suggestions indicated have been incorporated in the report. The course project report has been approved as it satisfies the academic requirements prescribed for the said degree.

Date:

Place: Belagavi

Signature of guide

Prof. Padma Dandannavar

Asst., Prof., Dept of CSE

KLS Gogte Institute Of Technology, Belagavi

Academic Year 2021 - 2022 (Even semester)

Course Seminar

Marks allocation:

	Batch No.: 3					
1.	Seminar Title: Database for a Pharmacy	Marks Range	**USN**			
			2GI20CS134	**2GI20CS144**	**2GI20CS158**	**2GI20CS184**
2.	Abstract (PO2)	0-2				
3.	Application of the topic to the course (PO2)	0-3				
4.	Literature survey and its findings (PO2)	0-4				
5.	Methodology, Results and Conclusion (PO1, PO3, PO4)	0-6				
6.	Report and Oral presentation skill (PO9, PO10)	0-5				
	Total	20				

Course Project

Marks Allocation

	Batch No. : 3					
1.	Project Title: Database for a Pharmacy	Marks Range	USN			
2.	Problem statement (PO2)	0-1				
3.	Need Analysis, Variables involved (PO1,PO2)	0-2				
4.	Alternate solutions to solve the problem(PO3)	0-3				
5.	Comparison between the solutions and reasonfor selecting the final solution(PO1,PO3,PO4)	0-4				
6.	Working model of the final solution(PO3,PO12)	0-5				
7.	Report and Oral presentation skill (PO9,PO10)	0-5				
	Total	20				

Signature of Staff

ACKNOWLEDGEMENTS

We take this opportunity to express our gratitude to all those people who have been instrumental in making this project successful. We feel honoured to place warm salutation to KLS GOGTE INSTITUTE OF TECHNOLOGY, Belagavi, which gave the opportunity to study B.E. and strengthen our knowledge base. We would like to express sincere thanks to Dr. J. K. Kittur, Principal, G.I. T, Belagavi for his warm support throughout the B.E. program. We are extremely thankful to Dr. Vijay S Rajpurohit, professor &Head Dept of CSE, G.I.T, Belagavi, for his constant co-operation and support throughout this project. We hereby express our thanks to Dr.Padma Dandannavar, Dept of CSE ,G.I.T, Belagavi for being guide for this project. She has provided us incessant support and has been a constant source of inspiration throughout the project. We thank all the teaching, Non- Teaching and Technical staff of Computer Science and Engineering Department, K.L.S. GOGTE INSTITUTE OF TECHNOLOGY, Belagavi for their invaluable support and guidance.

Shamant Myageri 2GI20CS1

Shradha Mallikarjun Patil 2GI20CS144

Srushti Mudennavar 2GI20CS158

Yash Herekar 2GI20CS184

ABSTRACT

Pharmacists can use the Pharmacy Management System program to help them methodically manage their pharmacies. When a medicine's name is input, the Pharmacy Management System can help by providing details about the medicine. A computer displays information about the medicine, such as its dosage and expiration date. In large medical stores, manually handling the specifics of all the drugs becomes very tough.

We can keep track of all the medicines by using this pharmacy management system. It is updated with new information as new medicines are introduced, and it includes an expiration date as well as a search option. When we complete the name of a medicine, it displays the medicine's details.

We can keep track of employee's details such as name, shift time and phone number. We also keep track of the drug manufacturer, warehouse, supplier and customer details.

This system was coded in python. The SQL database was created with MySQL.

CONTENTS

CHAPTER 1

INTRODUCTION

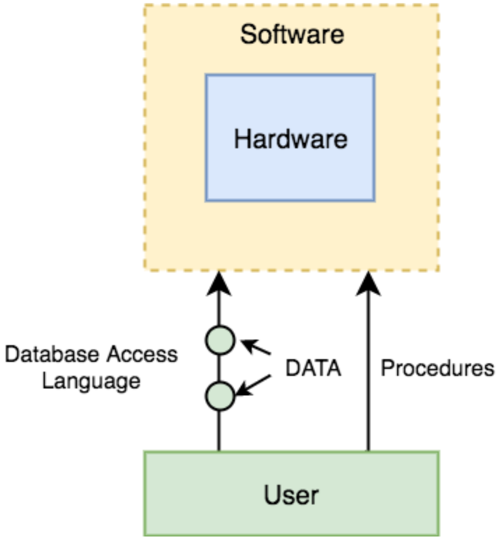
1.1 OVERVIEW

A database management system (DBMS) is a software package designed to define, manipulate, retrieve and manage data in a centralised database. A DBMS generally manipulates the data itself, the data format, field names, record structure and file structure. It also defines rules to validate and manipulate this data.

The earliest databases could only handle data formatted in a special way. Over time, the models for database management systems have changed considerably. This is a key part of understanding how various DBMS options work.

The relational model, which saves data in table formats, is the most widely used DBMS. The relational DBMS organises information into rows, columns, and tables, making it easier to find relevant information. Relational databases are popular because they are easy to extend, and new data categories can be added after the original database is created without large amounts of modification.

The Structured Query Language (SQL) is considered the standard user and application program interface for a relational database, and all relational DBMS software supports SQL. Examples include FileMaker Pro, Microsoft Access, Microsoft SQL Server, MySQL, and Oracle.



There are different types of databases. They are:

- Bibliographic
- full-text

- numeric
- images

In a database, even the smallest portion of information becomes the data. For example, a student is a data, a roll number is a data, and the address is data, height, weight, marks everything is data. In brief, all the living and non-living objects in this world are data.

1.2. PROBLEM STATEMENT:

PHARMACY MANAGEMENT SYSTEM

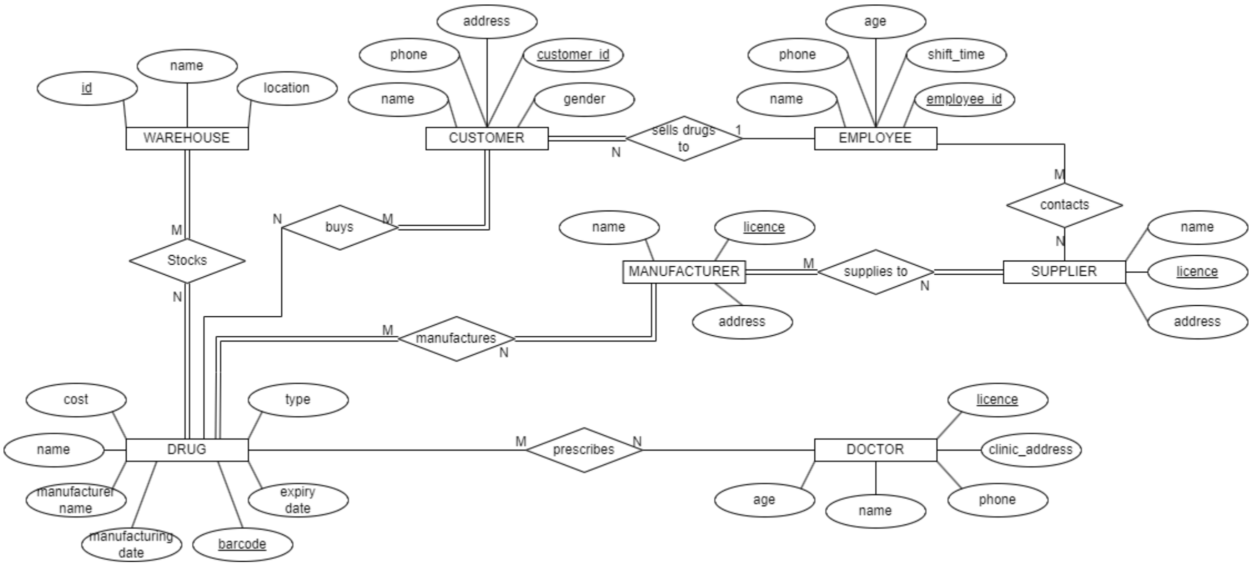
Most pharmacies face problems such as insufficient service promotions, lack of coherence of pharmacy services in hospitals, poor drug information systems, and the inconsistency of the pharmacy information management due to its manual processes.

The Pharmacy Management System is a project developed to automate medical stores' activities and improve their productivity. This helps pharmacies organise, manage, and secure drug information efficiently. Its' features aids in the resolution of challenges with manual pharmacy management.

A Pharmacy Management System can also help you keep track of your drug supplies. Prescriptions are proper and supplied in precise amounts using Pharmacy Management software. It oversees and manages the pharmacy team to preserve strong working relationships and outcomes. This can also improve quality and customer satisfaction ratings, as well as keep medicines from going bad.

CHAPTER 2:

ER DIAGRAM



2.1 ENTITIES

1. WAREHOUSE
2. EMPLOYEE
3. CUSTOMER
4. DRUG
5. DOCTOR
6. MANUFACTURER
7. SUPPLIER

2.2 RELATIONSHIPS

1. WAREHOUSE stores DRUG
2. EMPLOYEE contacts SUPPLIER
3. MANUFACTURER supplies SUPPLIER
4. CUSTOMER buys DRUG
5. DOCTOR prescribes DRUG
6. MANUFACTURER manufactures DRUG

2.3 ATTRIBUTES OF EACH ENTITY, RELATIONSHIP

1. WAREHOUSE: Name, Id, Location

2. EMPLOYEE: Name, Age, EmployeeId, PhoneNo, ShiftTime
3. CUSTOMER: Name, PhoneNo, Address, Gender, CustomerId
4. DRUG: Name, Cost, Type, ManufacturerName, ManufacturingDate, Barcode, Expiry Date
5. DOCTER: Name, Licence, PhoneNo , Age ,ClinicAddress
6. MANUFACTURER: Licence, Name, Address
7. SUPPLIER: Name, Licence, Address

2.4 CARDINALITY RATIO, PARTICIPATION RATIO

1. WAREHOUSE – DRUG (1: N) (Total- Partial)
2. EMPLOYEE – SUPPLIER (1: N) (Total- Partial)
3. CUSTOMER - DRUG (N: 1) (Total-Partial)
4. MANUFACTURER – SUPPLIER (M: N) (Total-Total)
5. DOCTOR – DRUG (M: N) (Partial- Partial)
6. MANUFACTURER – DRUG (N:M) (Total,Total)

CHAPTER 3:

SCHEMA

3.1 ER-TO-RELATIONAL MAPPING

Step 1: Mapping of Regular Entity Types

Step 2: Mapping of Weak Entity Types

Step 3: Mapping of Binary 1:1 Relation Types

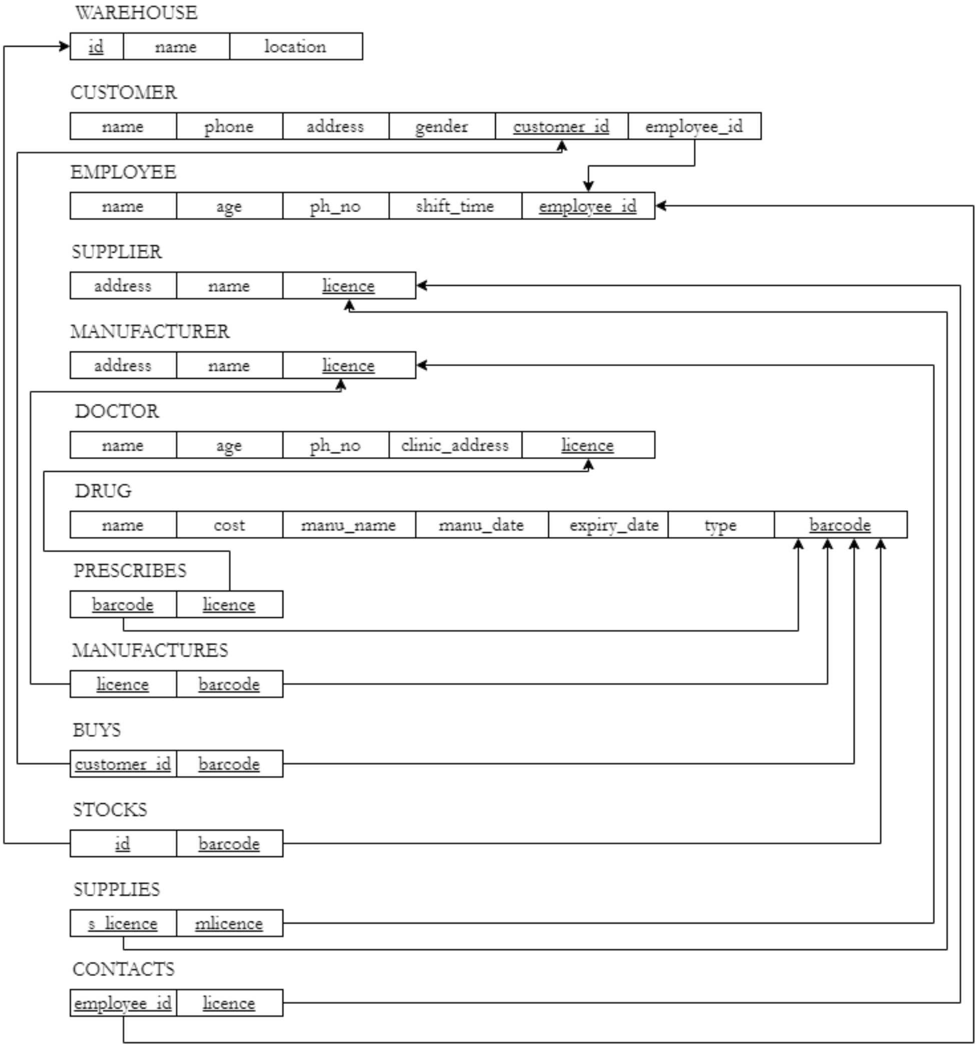
Step 4: Mapping of Binary 1:N Relationship Types.

Step 5: Mapping of Binary M:N Relationship Types.

Step 6: Mapping of Multivalued attributes.

Step 7: Mapping of N-ary Relationship Types.

3.2 SCHEMA DIAGRAM



WAREHOUSE: id | name | location

CUSTOMER: name | phone | address | gender | customer_id | employee_id

EMPLOYEE: name | age | ph_no | shift_time | employee_id

SUPPLIER: address | name | licence

MANUFACTURER: address | name | licence

DOCTOR: name | age | ph_no | clinic_address | licence

DRUG: name | cost | manu_name | manu_date | expiry_date | type | barcode

PRESCRIBES: barcode | licence

MANUFACTURES: licence | barcode

BUYS: customer_id | barcode

STOCKS: id | barcode

SUPPLIES: s_licence | mlicence

CONTACTS: employee_id | licence

CHAPTER 4:

SYSTEM REQUIREMENT SPECIFICATION

4.1 HARDWARE REQUIREMENTS:

1. Processor
2. Ram
3. HardDisks(SSD / HDD)
4. Network Adapter

4.2 SOFTWARE REQUIREMENTS

1. Operating System
2. MySQL Database Management Software
3. PyCharm IDE

CHAPTER 5:

IMPLEMENTATION

5.1 NORMALISATION

- Normalisation is the process of organising the data in the database.
- Normalisation is used to minimise the redundancy from a relation or set of relations. It is also used to eliminate undesirable characteristics like Insertion, Update, and Deletion Anomalies.
- Normalisation divides the larger table into smaller and links them using relationships.

1NF	A relation is in 1NF if it contains an atomic value.
2NF	A relation will be in 2NF if it is in 1NF and all non-key attributes are fully functional dependent on the primary key.
3NF	A relation will be in 3NF if it is in 2NF and no transition dependency exists.
BCNF	A stronger definition of 3NF is known as Boyce Codd's normal form.
4NF	A relation will be in 4NF if it is in Boyce Codd's normal form and has no multi-valued dependency.
5NF	A relation is in 5NF. If it is in 4NF and does not contain any join dependency, joining should be lossless.

The normal form is used to reduce redundancy from the database table.

Pharmacy database is in 2Nf

5.2 MODULES

CREATE TABLE STATEMENTS:

```
CREATE TABLE WAREHOUSE(
        ID INT,
        NAME VARCHAR(20),
        LOCATION VARCHAR(30),
        PRIMARY KEY(ID)
);

CREATE TABLE DRUG(
        NAME VARCHAR(30),
        COST REAL,
        TYPE VARCHAR(20),
        BARCODE INT,
        MANUFACTURING_DATE DATE,
        MANUFACTURER_NAME VARCHAR(20),
        EXPIRY_DATE DATE,
        PRIMARY KEY(BARCODE)
);

CREATE TABLE DOCTOR(
        NAME VARCHAR(20),
        AGE INT,
        PH_NO VARCHAR(10),
        LICENSE INT,
        CLINIC_ADDRESS VARCHAR(30),
        PRIMARY KEY(LICENSE)
);

CREATE TABLE MANUFACTURER(
        NAME VARCHAR(20),
        ADDRESS VARCHAR(30),
        LICENSE INT,
        PRIMARY KEY(LICENSE)
);

CREATE TABLE SUPPLIER(
        NAME VARCHAR(20),
        ADDRESS VARCHAR(30),
        LICENSE INT,
        PRIMARY KEY(LICENSE)
);
```

```
CREATE TABLE EMPLOYEE(
        NAME VARCHAR(20),
        AGE INT,
        PH_NO VARCHAR(10),
        EMPLOYEE_ID INT,
        SHIFT_TIME VARCHAR(20),
        PRIMARY KEY(EMPLOYEE_ID)
);

CREATE TABLE CUSTOMERS(
        NAME VARCHAR(20),
        PH_NO VARCHAR(10),
        ADDRESS VARCHAR(30),
        CUSTOMER_ID INT,
        GENDER VARCHAR(1),
        EMPLOYEE_ID INT,
        PRIMARY KEY(CUSTOMER_ID),
        FOREIGN KEY(EMPLOYEE_ID) REFERENCES
EMPLOYEE(EMPLOYEE_ID) ON DELETE SET NULL
);

CREATE TABLE STOCKS(
        BARCODE INT,
        ID INT,
        PRIMARY KEY(BARCODE,ID),
        FOREIGN KEY(BARCODE) REFERENCES DRUG(BARCODE) ON DELETE
CASCADE,
        FOREIGN KEY(ID) REFERENCES WAREHOUSE(ID) ON DELETE
CASCADE
);



CREATE TABLE BUYS(
        BARCODE INT,
        CUSTOMER_ID INT,
        PRIMARY KEY(BARCODE,CUSTOMER_ID),
	FOREIGN KEY(CUSTOMER_ID) REFERENCES
CUSTOMERS(CUSTOMER_ID) ON DELETE CASCADE,
	FOREIGN KEY(BARCODE) REFERENCES DRUG(BARCODE) ON DELETE
CASCADE
);

CREATE TABLE MANUFACTURES(
	BARCODE INT,
	MLICENSE INT,
	PRIMARY KEY(BARCODE,MLICENSE),
	FOREIGN KEY(BARCODE) REFERENCES DRUG(BARCODE) ON DELETE
CASCADE,
	FOREIGN KEY(MLICENSE) REFERENCES MANUFACTURER(LICENSE)
ON DELETE CASCADE
);

CREATE TABLE PRESCRIBES(
	BARCODE INT,
	DLICENSE INT,
	PRIMARY KEY(BARCODE,DLICENSE),
	FOREIGN KEY(BARCODE) REFERENCES DRUG(BARCODE) ON DELETE
CASCADE,
	FOREIGN KEY(DLICENSE) REFERENCES DOCTOR(LICENSE) ON
DELETE CASCADE
);

CREATE TABLE SUPPLIES(
	MLICENSE INT,
	SLICENSE INT,
	PRIMARY KEY(MLICENSE,SLICENSE),
	FOREIGN KEY(MLICENSE) REFERENCES MANUFACTURER(LICENSE)
ON DELETE CASCADE,
	FOREIGN KEY(SLICENSE) REFERENCES SUPPLIER(LICENSE) ON
DELETE CASCADE
);
```

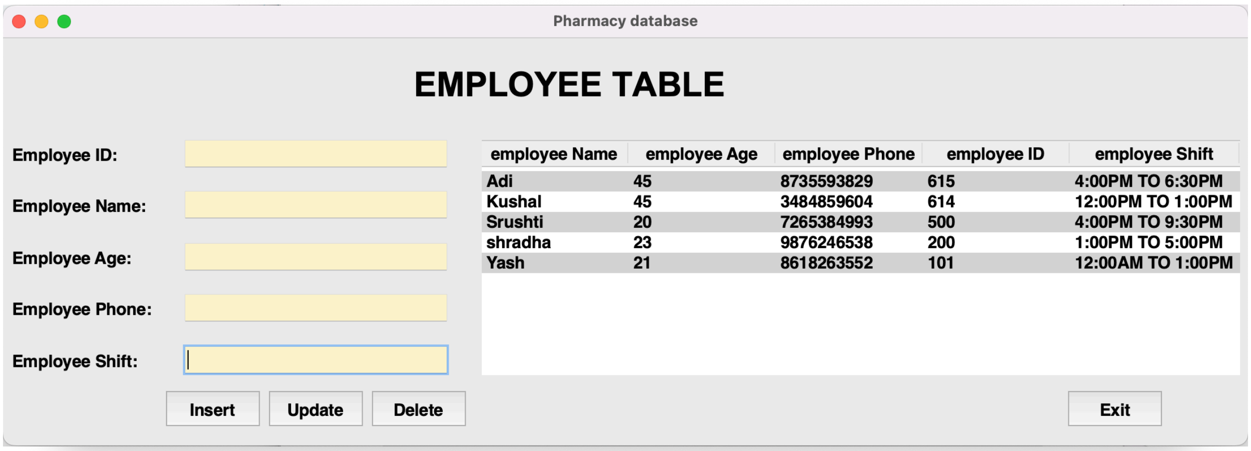
```
CREATE TABLE CONTACTS(
        EMPLOYEE_ID INT,
        SLICENSE INT,
        PRIMARY KEY(EMPLOYEE_ID,SLICENSE),
        FOREIGN KEY(EMPLOYEE_ID) REFERENCES
EMPLOYEE(EMPLOYEE_ID) ON DELETE CASCADE,
        FOREIGN KEY(SLICENSE) REFERENCES SUPPLIER(LICENSE) ON
DELETE CASCADE
);
#INSERT STATEMENTS:
INSERT INTO EMPLOYEE
VALUES('&NAME',&AGE,'&PH_NO',&EMPLOYEE_ID,'&SHIFT_TIME');
INSERT INTO CUSTOMERS
VALUES('&NAME','&PH_NO','&ADDRESS',&CUSTOMER_ID,'&GENDER',&EMP
LOYEE_ID);
INSERT INTO SUPPLIER VALUES('&NAME','&ADDRESS',&LICENSE);
INSERT INTO MANUFACTURER VALUES('&NAME','&ADDRESS',&LICENSE);
INSERT INTO DOCTOR
VALUES('&NAME',&AGE,'&PH_NO',&LICENSE,'&CLINIC_ADDRESS');
INSERT INTO DRUG
VALUES('&NAME',&COST,'&TYPE',&BARCODE,'&MANUFACTURING_DATE','&
MANUFACTURER_NAME','&EXPIRY_DATE');
INSERT INTO WAREHOUSE VALUES(&ID,'&NAME','&LOCATION');
INSERT INTO STOCKS VALUES(&BARCODE,&ID);
INSERT INTO BUYS VALUES(&BARCODE,&CUSTOMER_ID);
INSERT INTO MANUFACTURES VALUES(&BARCODE,&MLICENSE);
INSERT INTO PRESCRIBES VALUES(&BARCODE,&DLICENSE);
INSERT INTO SUPPLIES VALUES(&MLICENSE,&SLICENSE);
INSERT INTO CONTACTS VALUES(&EMPLOYEE_ID,&SLICENSE);
```

CHAPTER 6

RESULTS

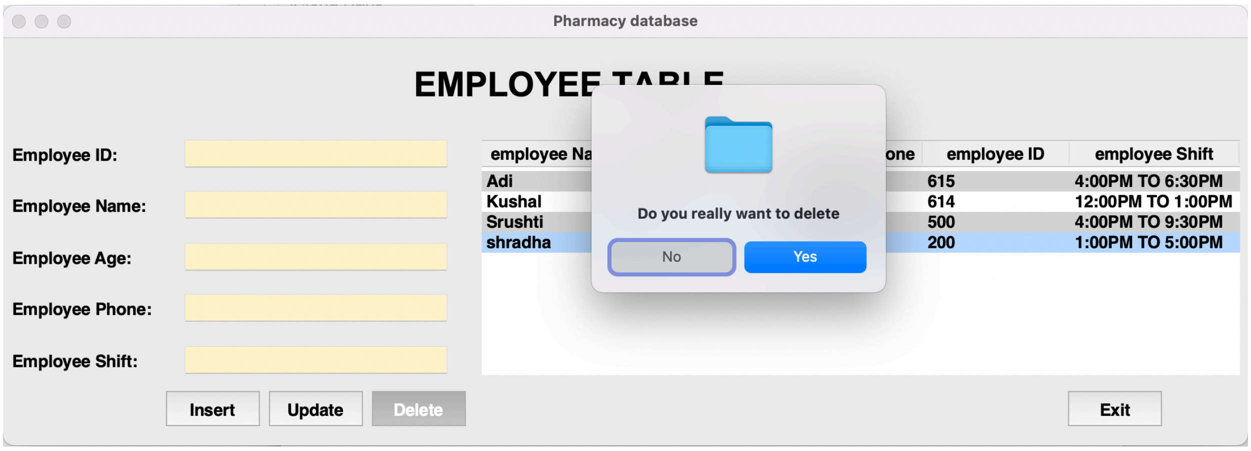
6.1 SCREENSHOTS

GUI



employee Name	employee Age	employee Phone	employee ID	employee Shift
Adi	45	8735593829	615	4:00PM TO 6:30PM
Kushal	45	3484859604	614	12:00PM TO 1:00PM
Srushti	20	7265384993	500	4:00PM TO 9:30PM
shradha	23	9876246538	200	1:00PM TO 5:00PM
Yash	21	8618263552	101	12:00AM TO 1:00PM

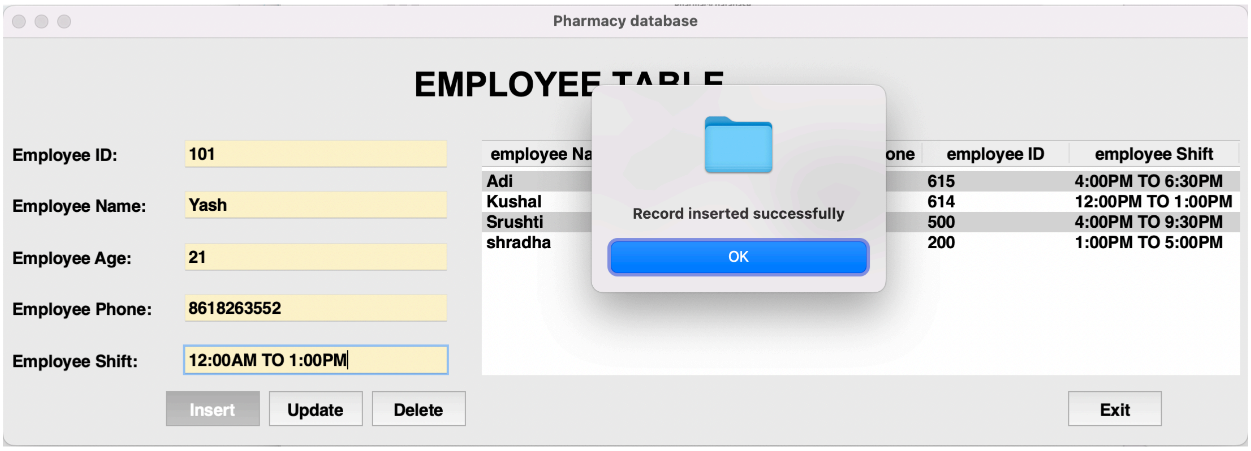
DELETE



Do you really want to delete

employee Na	one	employee ID	employee Shift
Adi		615	4:00PM TO 6:30PM
Kushal		614	12:00PM TO 1:00PM
Srushti		500	4:00PM TO 9:30PM
shradha		200	1:00PM TO 5:00PM

INSERTION



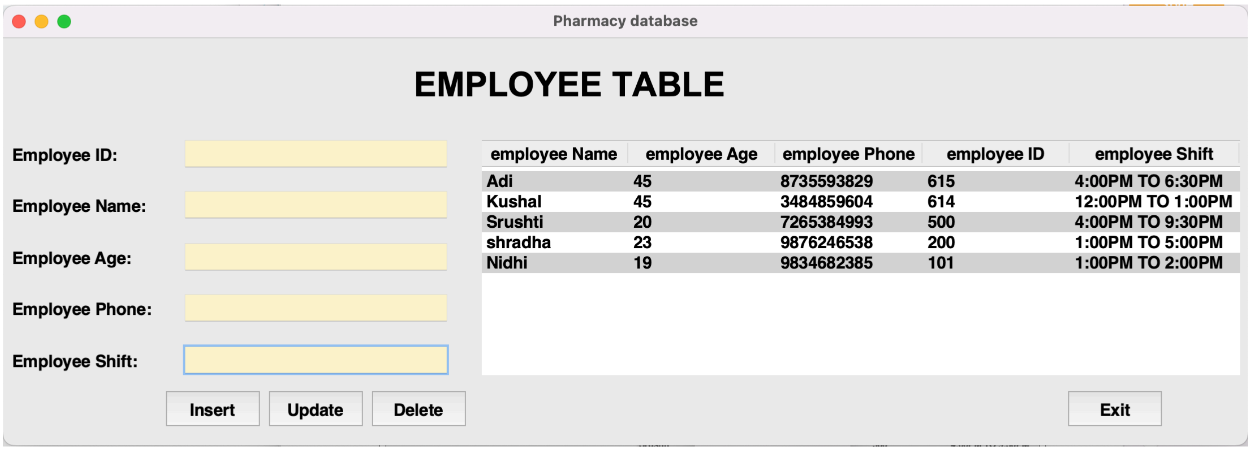
EMPLOYEE TABLE

Employee ID:	101
Employee Name:	Yash
Employee Age:	21
Employee Phone:	8618263552
Employee Shift:	12:00AM TO 1:00PM

employee Na	one	employee ID	employee Shift
Adi		615	4:00PM TO 6:30PM
Kushal		614	12:00PM TO 1:00PM
Srushti		500	4:00PM TO 9:30PM
shradha		200	1:00PM TO 5:00PM

Record inserted successfully

UPDATE

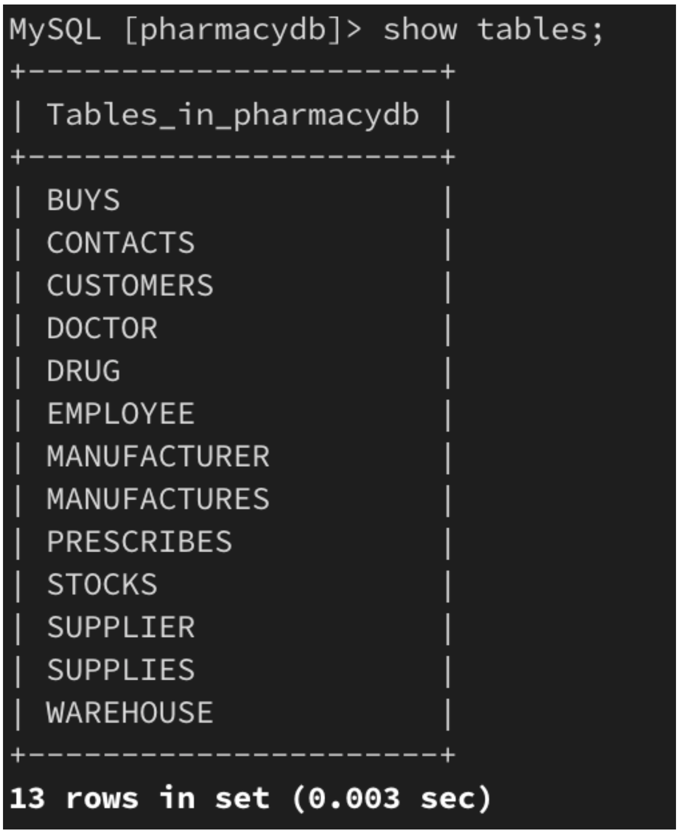


EMPLOYEE TABLE

Employee ID:	
Employee Name:	
Employee Age:	
Employee Phone:	
Employee Shift:	

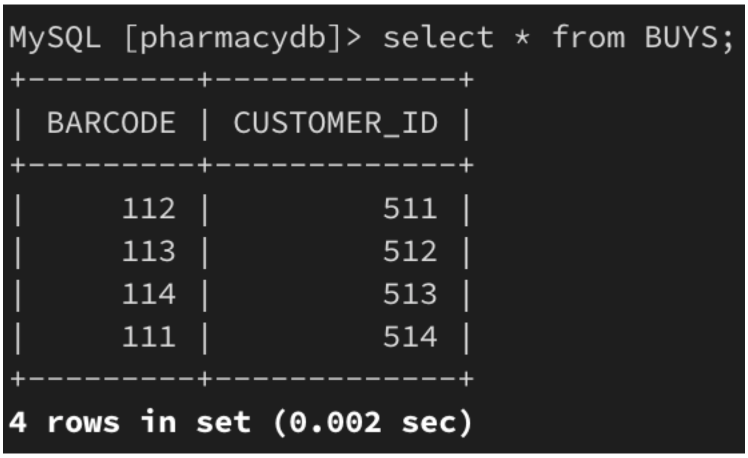
employee Name	employee Age	employee Phone	employee ID	employee Shift
Adi	45	8735593829	615	4:00PM TO 6:30PM
Kushal	45	3484859604	614	12:00PM TO 1:00PM
Srushti	20	7265384993	500	4:00PM TO 9:30PM
shradha	23	9876246538	200	1:00PM TO 5:00PM
Nidhi	19	9834682385	101	1:00PM TO 2:00PM

TABLES



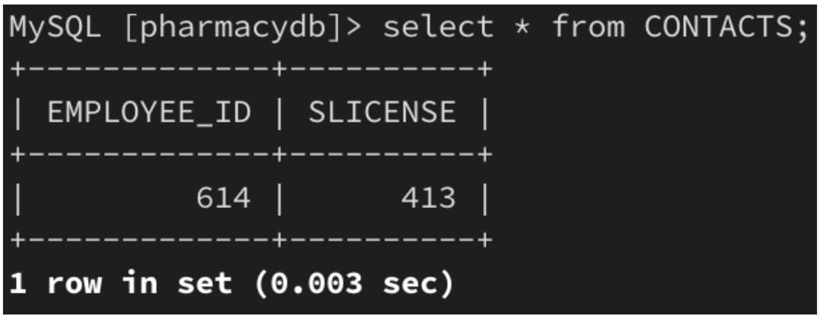
```
MySQL [pharmacydb]> show tables;
+----------------------+
| Tables_in_pharmacydb |
+----------------------+
| BUYS                 |
| CONTACTS             |
| CUSTOMERS            |
| DOCTOR               |
| DRUG                 |
| EMPLOYEE             |
| MANUFACTURER         |
| MANUFACTURES         |
| PRESCRIBES           |
| STOCKS               |
| SUPPLIER             |
| SUPPLIES             |
| WAREHOUSE            |
+----------------------+
13 rows in set (0.003 sec)
```

BUYS



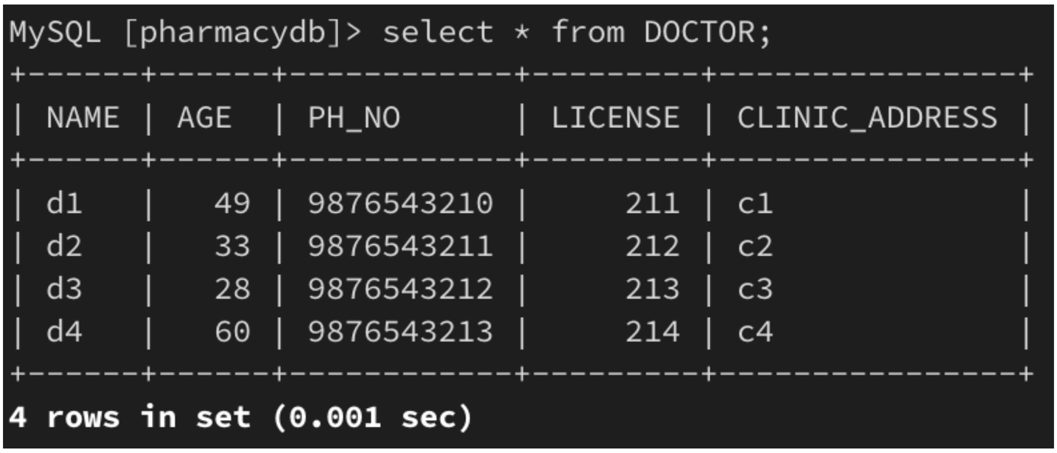
```
MySQL [pharmacydb]> select * from BUYS;
+---------+-------------+
| BARCODE | CUSTOMER_ID |
+---------+-------------+
|     112 |         511 |
|     113 |         512 |
|     114 |         513 |
|     111 |         514 |
+---------+-------------+
4 rows in set (0.002 sec)
```

CONTACTS



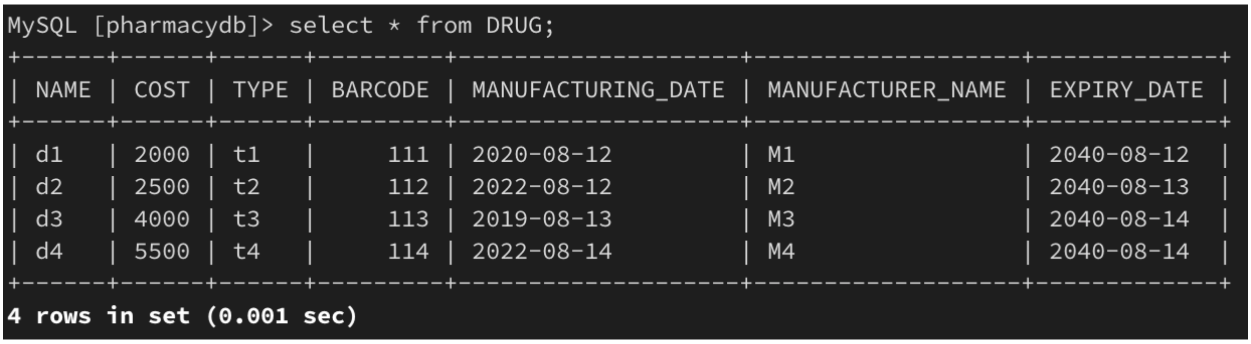
```
MySQL [pharmacydb]> select * from CONTACTS;
+-------------+----------+
| EMPLOYEE_ID | SLICENSE |
+-------------+----------+
|         614 |      413 |
+-------------+----------+
1 row in set (0.003 sec)
```

DOCTOR



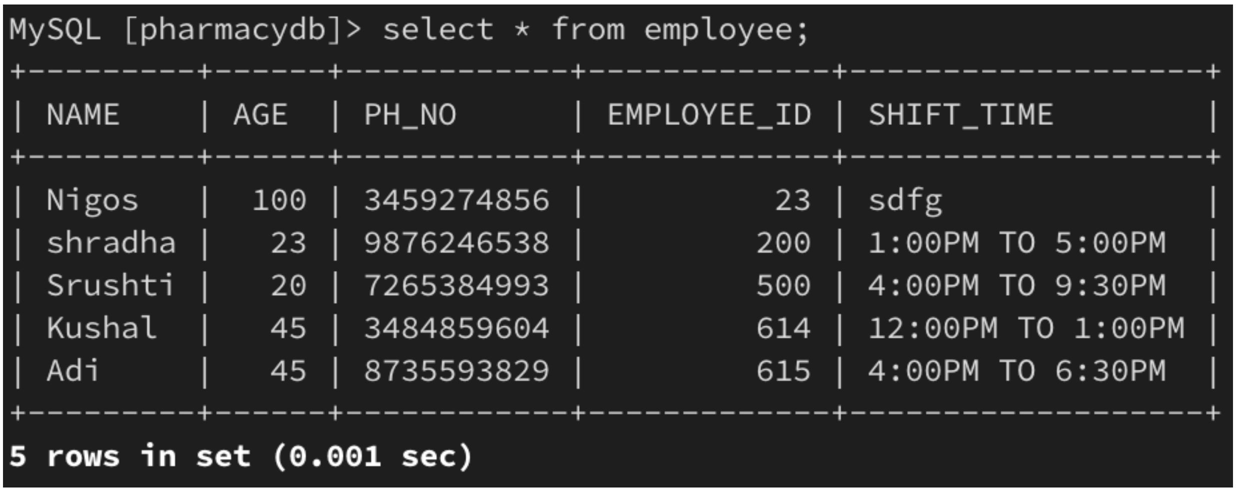
```
MySQL [pharmacydb]> select * from DOCTOR;
+------+------+------------+---------+----------------+
| NAME | AGE  | PH_NO      | LICENSE | CLINIC_ADDRESS |
+------+------+------------+---------+----------------+
| d1   |   49 | 9876543210 |     211 | c1             |
| d2   |   33 | 9876543211 |     212 | c2             |
| d3   |   28 | 9876543212 |     213 | c3             |
| d4   |   60 | 9876543213 |     214 | c4             |
+------+------+------------+---------+----------------+
4 rows in set (0.001 sec)
```

DRUG



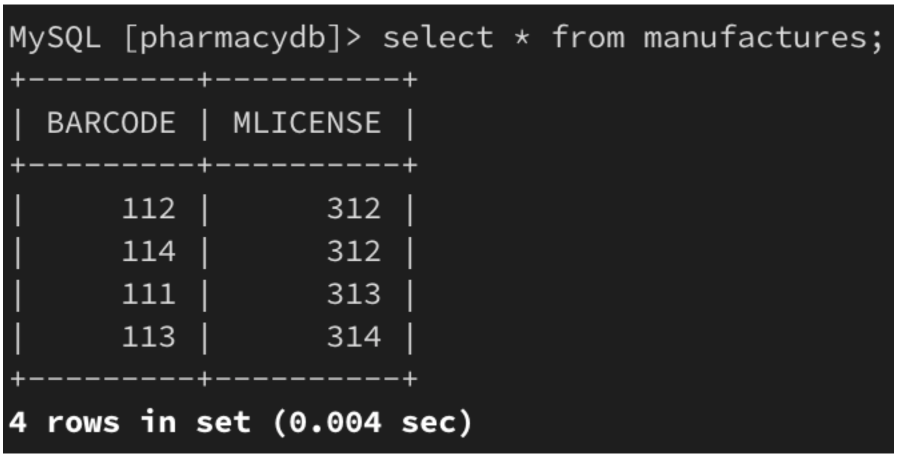
```
MySQL [pharmacydb]> select * from DRUG;
+------+------+------+---------+--------------------+-------------------+-------------+
| NAME | COST | TYPE | BARCODE | MANUFACTURING_DATE | MANUFACTURER_NAME | EXPIRY_DATE |
+------+------+------+---------+--------------------+-------------------+-------------+
| d1   | 2000 | t1   |     111 | 2020-08-12         | M1                | 2040-08-12  |
| d2   | 2500 | t2   |     112 | 2022-08-12         | M2                | 2040-08-13  |
| d3   | 4000 | t3   |     113 | 2019-08-13         | M3                | 2040-08-14  |
| d4   | 5500 | t4   |     114 | 2022-08-14         | M4                | 2040-08-14  |
+------+------+------+---------+--------------------+-------------------+-------------+
4 rows in set (0.001 sec)
```

EMPLOYEE



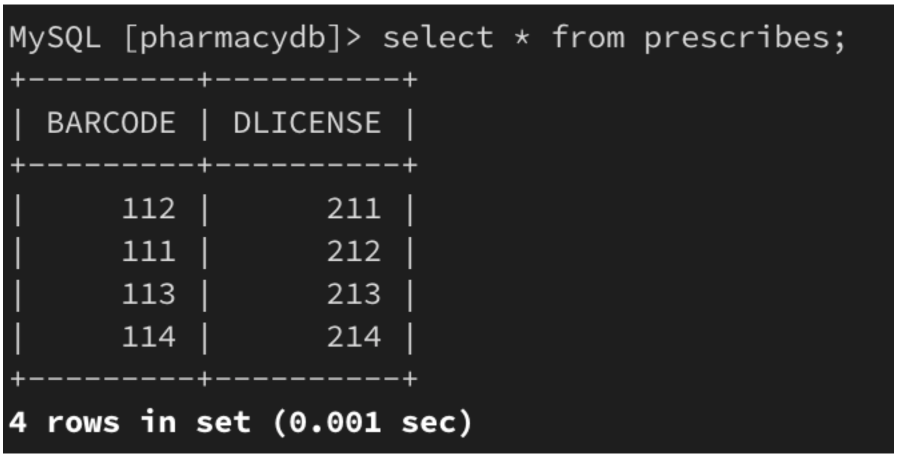
```
MySQL [pharmacydb]> select * from employee;
+---------+------+------------+-------------+------------------+
| NAME    | AGE  | PH_NO      | EMPLOYEE_ID | SHIFT_TIME       |
+---------+------+------------+-------------+------------------+
| Nigos   |  100 | 3459274856 |          23 | sdfg             |
| shradha |   23 | 9876246538 |         200 | 1:00PM TO 5:00PM  |
| Srushti |   20 | 7265384993 |         500 | 4:00PM TO 9:30PM  |
| Kushal  |   45 | 3484859604 |         614 | 12:00PM TO 1:00PM |
| Adi     |   45 | 8735593829 |         615 | 4:00PM TO 6:30PM  |
+---------+------+------------+-------------+------------------+
5 rows in set (0.001 sec)
```

MANUFACTURES



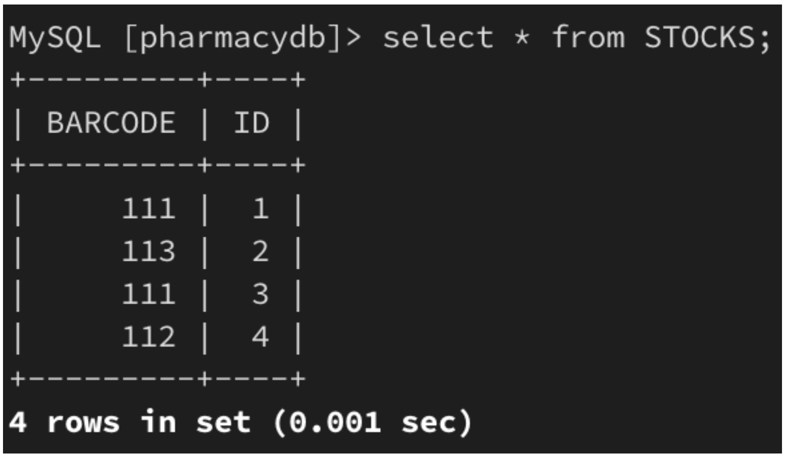
```
MySQL [pharmacydb]> select * from manufactures;
+---------+----------+
| BARCODE | MLICENSE |
+---------+----------+
|     112 |      312 |
|     114 |      312 |
|     111 |      313 |
|     113 |      314 |
+---------+----------+
4 rows in set (0.004 sec)
```

PRESCRIBES



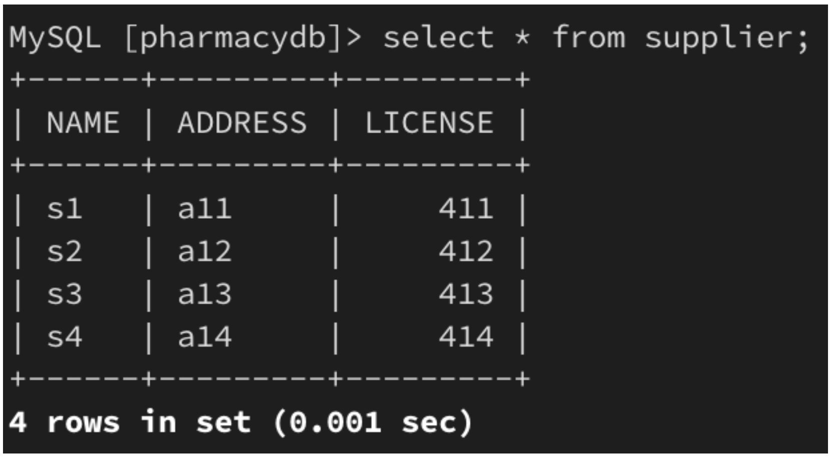
```
MySQL [pharmacydb]> select * from prescribes;
+---------+----------+
| BARCODE | DLICENSE |
+---------+----------+
|     112 |      211 |
|     111 |      212 |
|     113 |      213 |
|     114 |      214 |
+---------+----------+
4 rows in set (0.001 sec)
```

STOCKS



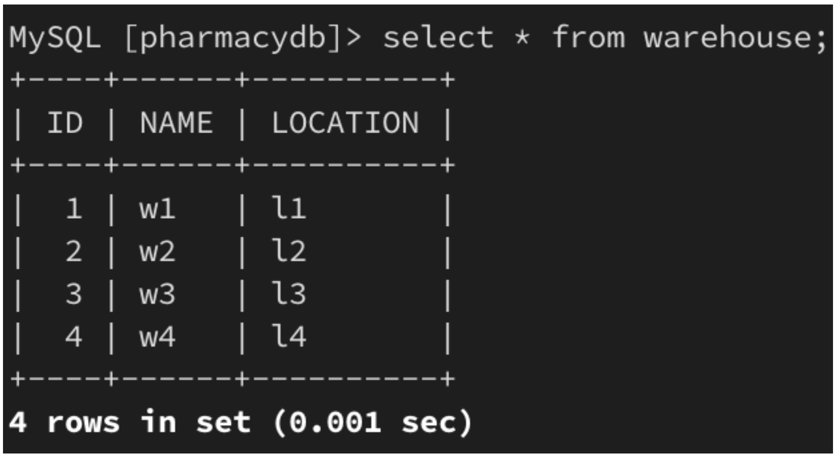
```
MySQL [pharmacydb]> select * from STOCKS;
+---------+----+
| BARCODE | ID |
+---------+----+
|     111 |  1 |
|     113 |  2 |
|     111 |  3 |
|     112 |  4 |
+---------+----+
4 rows in set (0.001 sec)
```

SUPPLIER



```
MySQL [pharmacydb]> select * from supplier;
+------+---------+---------+
| NAME | ADDRESS | LICENSE |
+------+---------+---------+
| s1   | a11     |     411 |
| s2   | a12     |     412 |
| s3   | a13     |     413 |
| s4   | a14     |     414 |
+------+---------+---------+
4 rows in set (0.001 sec)
```

WAREHOUSE



```
MySQL [pharmacydb]> select * from warehouse;
+----+------+----------+
| ID | NAME | LOCATION |
+----+------+----------+
|  1 | w1   | l1       |
|  2 | w2   | l2       |
|  3 | w3   | l3       |
|  4 | w4   | l4       |
+----+------+----------+
4 rows in set (0.001 sec)
```

CONCLUSION

In this course project we learnt how to use open-source ER design tool diagram and created the ER model for the problem statement. Converted the ER diagram to relational Schema by applying the relational rules. We identified the primary and foreign key constraints and created all the relations in sqlite3 database using the DDL statements. Further we populated each of the tables with data using the insert statement. We also learnt the use of basic SQL statements like insert, delete, update and select to update or alter the table and also display the table data. We learnt how to build a GUI using python and connect edit to the sqlite database. Lastly, we learnt how to perform all basic operations and queries using GUI application and also reflect the data in the original database.

REFERENCES

[1] https://www.altexsoft.com/blog/pharmacy-management-system/

[2] Fundamentals of Database Systems, 6th Edition