**Pharmaceutical Management System**

**Supervisor**

**Basit Raza**

**By:**

**Muhammad Usman SP20-BSE-074**

**Muhammad Tahir Hassan Khan SP20-BSE-073**

**Umair SP20-BSE-096**

PROJECT REPORT

2021

Contents

[2 Abstract: 1](#_Toc90450312)

[3 Introduction: 1](#_Toc90450313)

[4 Problem Statement: 1](#_Toc90450314)

[5 Problem Solution/Objectives of the Proposed System: 1](#_Toc90450315)

[6 Vision Statement: 2](#_Toc90450316)

[7 Scope: 2](#_Toc90450317)

[8 System Limitations/Constraints: 2](#_Toc90450318)

[9 Tools and Technologies: 2](#_Toc90450319)

[10 Project Stakeholders and Roles: 3](#_Toc90450320)

[11 Gantt Chart: 3](#_Toc90450321)

[12 Table Descriptions: 3](#_Toc90450322)

[13 Data Model: 5](#_Toc90450323)

[14 Functional Dependencies and Normalization: 7](#_Toc90450324)

[15 Indexing: 9](#_Toc90450325)

[16 Data Definition Language (DDL) commands 11](#_Toc90450326)

[17 Tables: 18](#_Toc90450327)

[18 SQL QUERIES: 22](#_Toc90450328)

# Abstract:

One of the most important responsibilities of pharmacy management is to supervise and manage the pharmacy employees to ensure healthy working relationships and outcomes. Each of these functions is critical to the pharmacy’s operation and should be explained by the management.

However, most pharmacies faced 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. Now these are the problems that must be solved with this Pharmacy Management System Project Proposal.

# Introduction:

In general, Pharmaceutical Management/Information is a system that stores data and enables functionality that organizes and maintains the medication use process within pharmacies. These systems may be an independent technology for the pharmacy's use only, or in a hospital setting, pharmacies may be integrated within an inpatient hospital system. The mini-world I have chosen is a Pharmacy-Retail Management System in a hospital setting.

# Problem Statement:

First, selecting the suitable medicine for the type of illness is usually take time and makes the patient or customer waiting. Therefore, the time is waste for the customer to be waiting. Second, for the medicines stock management, the pharmacist must check it manually and no warning message for the medicine that decrease to finish.

# Problem Solution/Objectives of the Proposed System:

So, PMS provide the advantage to setup the process in selecting and alert program to warn about the medicine stock.

Finally, yet important, no analyses are done for the frequency type of medicine usually buy by the customer or patient at that area. This is also important to determine the medicines that are demanded more from the customers so that pharmacist can be prepared to order for that type of medicine.

**Objectives:**

The objective to be achieved from this PMS is:

* To care the pharmacist alert about the out-of-stock medicine.
* To provide the reliable medicine selector for the minor illness.
* To provide a secure system in terms of information retravel.
* To provide a systematic medicine inventory.

# Vision Statement:

This Pharmaceutical - Retail Management system works well when concentrated towards the retail sector of buying and selling. It also works well with the Hospital side of the Pharmacy which involves the whole process of the patient getting treated in a hospital by many doctors. This database system provides essential daily store transactions statistics. The record of medicine along its dealer selling to a particular store can also be kept. The contract between the store and hospital along with details of each patient, hospital, store, dealer, and medicine are also stored.

# Scope:

In general, Pharmaceutical Management/Information is a system that stores data and enables functionality that organizes and maintains the medication use process within pharmacies. These systems may be an independent technology for the pharmacy's use only, or in a hospital setting, pharmacies may be integrated within an inpatient hospital system. The mini-world we have chosen is a Pharmacy-Retail Management System in a hospital setting.

# System Limitations/Constraints:

When it comes to limitations if a PATIENT buys 2 or more different medicines in the same store in the same day, then the bill\_id, pat\_id, store\_id has to be the same or else the output doesn't make sense (Eg: bill\_id is same but store\_id is different on the same purchase date!). CONTRACTS are assumed to be that 1 STORE can have contracts with many HOSPITALS and not the other way around, so that it focuses on the importance of each store. treat\_id, hosp\_id, pat\_id, store\_id, contract\_id, dealer\_id and med\_id are assumed to be unique in their respective tables. It is assumed that a doctor can only work in 1 hospital and multiple doctors can work in 1 hospital. Before transacting, it has to be taken care that the corresponding quantity of tablets are available in the store.

# Tools and Technologies:

Tools and Technologies for Proposed Project

|  |  |  |  |
| --- | --- | --- | --- |
| **Tools**  **And**  **Technologies** | **Tools** | **Version** | **Rationale** |
| MS Visual Studio | 2015 | IDE |
| MS SQL Server | 2015 | DBMS |
| Adobe Photoshop | CSC 6 | Design Work |
| **Technology** | **Version** | **Rationale** |
| Java | 8.0.12 | Front-end Development |
| SQL Developer | 21.2.1.204.1703 | Back-end Development |

# Project Stakeholders and Roles:

Project Stakeholders for Proposed Project

|  |  |
| --- | --- |
| **Project Sponsor** | COMSATS University Islamabad, Islamabad Campus |
| **Stakeholder** | Muhammad Usman, Muhammad Tahir Hassan Khan, Umair   * Basit Raza |

# Gantt Chart:

***Graphical user interface

Description automatically generated***

# Table Descriptions:

**1. MEDICINE:**

Keep information about medicine link medicine id(med\_id), name, composition, manufacture and expiry date(mfg\_date, exp\_date) and cost per tablet(cost\_per\_tab). Each store has certain available tablets in stock, supplied by one or more dealers.

**2. DEALER:**

Holds details about the Dealer id(dealer\_id), name, address, phone. - Dealers supply medicines to the stores.

**3. HOSPITAL:**

Holds information about the hospital id(hos\_id), name, address and phone. - Patients are treated at the hospitals. - Hospitals have contracts with one or many stores.

**4. DOCTOR:**

Contains information like doctor id(hos\_id), hospital is(hos\_id) and doctor name(doc\_name). - hos\_id indicates the hospital in which the doctor works.

**5. PATIENT:**

Keeps information like patient id(pat\_id), name, address and phone. - Patient is strictly treated at a hospital. - Is part of the TRANSACTION, to buy medicines in stores.

**6. TREATMENT:**

Indicates the attributes of a relationship mainly, the patient involved(pat\_id), the hospital in which he was treated(hos\_id), the doctor, in that hospital, he was treated under(doc\_id) and the treatment date.

**7. TRANSACTIONS:**

It is a process that involves the patient(pat\_id) buying medicines(med\_id) in few/bulk quantities(quantity) at a store(store\_id) generating a bill(bill\_id) and a total on a particular date(pur\_date). This entire process, between the PATIENT and the STORES, has been modeled into this single relation.

**8. RETAIL:**

It is a relationship between the dealer(dealer\_id) and the store(store\_id). The medicines(med\_id) and the batch number(batch\_no) along with quantity\_supplied is noted down.

**9. CONTRACT:**

It is a contract between the Hospital(hos\_id) and the Store(store\_id).

**10. STORES:**

Has information about the store(store\_id), name, address, phone and store manager(store\_man).

**11. Supplies:**

Has information about the supplies(supplies\_id), store(store\_id) and dealers(dealer\_id).

# Data Model:

Diagram

Description automatically generated**ER Diagram:**

**Fig 1: ER diagram of Pharmaceutical-Retail Database System**

* All the strong(regular) entities, in Black, have a corresponding relation of their own - MEDICINE, STORES, DEALER, PATIENT, DOCTOR and HOSPITAL.
* The entire process of a PATIENT transacting with a STORE and the same STORE generating a bill which is ISSUED back to the PATIENT is modeled into a single relation called TRANSACTION. The ERD elaborates what is involved in a TRANSACTION. The transactions are stored such that each tuple represents only a single medicine being bought, and there will be ‘n’ rows signifying a patient buying ‘n’ medicines.
* RETAIL - It is a relationship between the dealer(dealer\_id) and the store(store\_id). The medicines(med\_id) and the batch number(batch\_no) along with quantity\_supplied is noted down.
* CONTRACT - It is a contract between the HOSPITAL(hos\_id) and the STORE(store\_id). It had to be made into a separate table even though it has only two attributes, because the earlier design was not in 2 Normal Form (2NF).
* TREATMENT - here is modeled as a ternary relationship. For simplicity it can be viewed as a PATIENT consulting a DOCTOR and the DOCTOR providing treatment in a HOSPITAL.

**Relational Schema:**

Diagram

Description automatically generatedEach relation is mapped using an appropriate foreign key keeping in mind the cardinality and participation constraints.

**Fig 2: Relational Schema of Pharmaceutical-Retail Database System**

# Functional Dependencies and Normalization:

|  |  |  |  |
| --- | --- | --- | --- |
| **Relation** | **Functional Dependencies** | **Primary Key** | **Candidate Keys** |
| MEDICINE | med\_id → {name, composition,  mfg\_date, exp\_date, cost\_per\_tab}    (name, composition) → {cost\_per\_tab} | med\_id | med\_id |
| STORES | store\_id → {name, address, contact,  store\_man}    (name, address) →{store\_id, contact, store\_man} | store\_id | store\_id |
| DEALER | dealer\_id → {name, address, phone}    (name, address) → {dealer\_id, phone} | dealer\_id | dealer\_id |
| PATIENT | pat\_id → {name, address, phone}    (name, address) → {pat\_id, phone} | pat\_id | pat\_id |
| HOSPITAL | hos\_id → {name, address, phone}    (name, address) → {hos\_id, phone} | hos\_id | hos\_id |
| DOCTOR | doc\_id → { hos\_id, doc\_name}    (hos\_id, doc\_name) → doc\_id | doc\_id | doc\_id, hos\_id |
| TRANSACTIONS | bill\_id → {pat\_id, store\_id, med\_id, quantity, pur\_date, total}    (bill\_id, pat\_id, store\_id, med\_id) →  {quantity, pur\_date, total} | bill\_id | bill\_id, pat\_id, store\_id, med\_id |
| RETAIL | reatil\_id → {med\_id, store\_id,  dealer\_id, batch\_no, quantity\_supplied} | retail\_id | retail\_id, med\_id, store\_id, dealer\_id |
| TREATMENT | treat\_id → {pat\_id, hos\_id, doc\_id, treat\_date}    (treat\_id, pat\_id, hos\_id, doc\_id) → treat\_date | treat\_id | treat\_id, hos\_id, doc\_id, pat\_id |
| CONTRACT | contract\_id → {hos\_id, store\_id} | contract\_id | contract\_id, hos\_id, store\_id |
| QUANT | (med\_id, store\_id) → {quantity} |  | med\_id, store\_id |

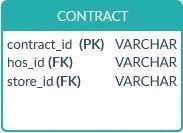
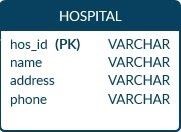
**First Normal Form (1NF):**

While converting an ER diagram to a Relational Mapping it is ensured that if any multivalued attributes are present then they are given separate independent tables. Since all the relations(tables) have a unique set of columns, with all the values in a particular column being atomic and belonging to the same domain, the relations are in 1NF.

Graphical user interface, application

Description automatically generated**Second Normal Form (2NF):**

**Fig 3: HOSPITAL relation not in 2NF**



**Fig 4: Solution to partial dependency in Fig 3**

In the earlier design, given in Fig 3, the relation HOSPITAL had store\_id as an attribute under it to indicate the tie-up/contract that the hospital(hos\_id) had with the store(store\_id). But here, it can be seen that following dependency holds :

* *hos\_id → {store\_id, name, address, phone}*
* *hos\_id → {name, address, phone}*
* *(name, address) → {hos\_id, phone}*

But in no way does store\_id determine other attributes of HOSPITAL excluding hos\_id i.e *store\_id →*​ {name, address, phone} is false.

Thus *{*​ *name, address, phone}*​ depend only on a proper subset of prime attributes i.e only *hos\_id*​ .Thus there exists a partial dependency here.

Fig 4: Shows how the partial dependency was taken care off by utilizing an extra table called CONTRACT. Now *hos\_id → {name, address, phone}*​ ​ and *(*​ *name, address) → {hos\_id, phone}*​. In CONTRACT table *contract\_id → {hos\_id, store\_id}*​ dependency holds.

Thus, by eliminating the partial dependency in the table HOSPITAL and since all the other relations are in 1NF, without any partial dependency in them, the overall schema is in 2NF.

Here, since hos\_id is the candidate key in either table which is an unique, primary key itself, the broken down tables have a lossless decomposition.

*attributes(CONTRACT)* ​U *attributes(HOSPITAL)*​ ​ = *attributes(HOSPITAL’’)*​ *attributes(CONTRACT)* ​∩ *attributes(HOSPITAL)* ​ ​ ≠ Φ

**Third Normal Form (3NF):**

Earlier in the schema design , Fig 5, shows the relation TREATMENT. <Excluded ‘*type*​ ​’ attribute later on>

Graphical user interface, text, application

Description automatically generated with medium confidence

**Fig 5: TREATMENT relation not in 3NF**



**Fig 6: Solution to transitive dependency in Fig 5**

The dependencies were *pat\_id*​ *→ {hos\_id}* and *hos\_id*​ *→ {doc\_name}*​. This shows true transitive dependency between *pat\_id*​ and *doc\_name*​ ​, which is not true at all. Thus, this table is not in 3NF. Hence, this was resolved using another table called DOCTOR to separately hold the contents of the doctor working in one or more hospitals/clinics. By doing so the TREATMENT table now is free from transitive dependency and since it is already in 1NF and 2NF, the relation is now in 3NF, and the decomposition is lossless.

Giving another example, if the TREATMENT relation had another attribute called hos\_name, indicating the name of the hospital where the treatment was carried out, this table would have transitive dependency because *treat\_id*​ *→ {hos\_id}* and *hos\_id*​ *→ {hos\_name}* would now come into existence and resulting in a 3NF violation.

Since all the relations are in 3NF, the schema relations are now normalised until the 3NF.

# Indexing:

Indexing is a way to optimize the performance of a database by minimizing the number of disk accesses required when a query is processed. It is a data structure technique which is used to quickly locate and access the data in a database. Indexes are created using a few database columns.

* The first column is the Search key that contains a copy of the primary key or candidate key of the table. These values are stored in sorted order so that the corresponding data can be accessed quickly.
* The second column is the Data Reference or Pointer which contains a set of pointers holding the address of the disk block where that key value can be found.

Table

Description automatically generated with medium confidenceWe have use indexing in our project on different attributes of different tables. As shown below all the index we used in project.

Here is example of indexing:

In this query we want the dealer whose id is d3:

Graphical user interface, text, application

Description automatically generated

# Data Definition Language (DDL) commands

CREATE TABLE contract (

hospital\_hosid VARCHAR2(100) NOT NULL,

stores\_store\_id VARCHAR2(100) NOT NULL,

contractid VARCHAR2(100) NOT NULL

);

CREATE UNIQUE INDEX contract\_\_idx ON

contract (

contractid

ASC );

ALTER TABLE contract ADD CONSTRAINT contract\_pk PRIMARY KEY ( hospital\_hosid,

stores\_store\_id );

CREATE TABLE dealer (

dealid VARCHAR2(100) NOT NULL,

name VARCHAR2(200) NOT NULL,

address VARCHAR2(200),

phoneno VARCHAR2(200) NOT NULL

);

CREATE INDEX dealer\_\_idxv1 ON

dealer (

dealid

ASC );

ALTER TABLE dealer ADD CONSTRAINT dealer\_pk PRIMARY KEY ( dealid );

ALTER TABLE dealer ADD CONSTRAINT dealer\_dealid\_un UNIQUE ( dealid );

CREATE TABLE doctor (

docid VARCHAR2(100) NOT NULL,

docname VARCHAR2(200) NOT NULL

);

CREATE UNIQUE INDEX doctor\_\_idx ON

doctor (

docid

ASC );

ALTER TABLE doctor ADD CONSTRAINT doctor\_pk PRIMARY KEY ( docid );

CREATE TABLE hospital (

hosid VARCHAR2(100) NOT NULL,

name VARCHAR2(200),

address VARCHAR2(200) NOT NULL,

phoneno VARCHAR2(200)

);

CREATE UNIQUE INDEX hospital\_\_idx ON

hospital (

hosid

ASC );

ALTER TABLE hospital ADD CONSTRAINT hospital\_pk PRIMARY KEY ( hosid );

CREATE TABLE medicine (

med\_id VARCHAR2(100) NOT NULL,

name VARCHAR2(200) NOT NULL,

mfg\_date DATE,

exp\_date DATE,

composition VARCHAR2(200),

cost\_per\_tab INTEGER

);

COMMENT ON COLUMN medicine.exp\_date IS

' ';

CREATE UNIQUE INDEX medicine\_\_idx ON

medicine (

med\_id

ASC );

ALTER TABLE medicine ADD CONSTRAINT medicine\_pk PRIMARY KEY ( med\_id );

CREATE TABLE patient (

patid VARCHAR2(100) NOT NULL,

name VARCHAR2(200) NOT NULL,

address VARCHAR2(200),

phoneno VARCHAR2(200) NOT NULL

);

CREATE UNIQUE INDEX patient\_\_idx ON

patient (

patid

ASC );

ALTER TABLE patient ADD CONSTRAINT patient\_pk PRIMARY KEY ( patid );

CREATE TABLE retail (

medicine\_med\_id VARCHAR2(100) NOT NULL,

stores\_store\_id VARCHAR2(100) NOT NULL,

retail\_id VARCHAR2(100) NOT NULL,

batchno INTEGER,

quantity INTEGER

);

CREATE UNIQUE INDEX retail\_\_idx ON

retail (

medicine\_med\_id

ASC,

stores\_store\_id

ASC );

ALTER TABLE retail ADD CONSTRAINT retail\_pk PRIMARY KEY ( medicine\_med\_id,

stores\_store\_id );

CREATE TABLE stores (

store\_id VARCHAR2(100) NOT NULL,

name VARCHAR2(200),

manager VARCHAR2(200),

address VARCHAR2(200),

contact\_no VARCHAR2(200)

);

CREATE UNIQUE INDEX stores\_\_idx ON

stores (

store\_id

ASC );

ALTER TABLE stores ADD CONSTRAINT stores\_pk PRIMARY KEY ( store\_id );

CREATE TABLE supplies (

stores\_store\_id VARCHAR2(100) NOT NULL,

suppliesid VARCHAR2(100) NOT NULL,

dealer\_dealid12 VARCHAR2(100) NOT NULL,

dealer\_dealid1 VARCHAR2(200) NOT NULL

);

CREATE UNIQUE INDEX supplies\_\_idx ON

supplies (

stores\_store\_id

ASC );

ALTER TABLE supplies

ADD CONSTRAINT supplies\_pk PRIMARY KEY ( suppliesid,

dealer\_dealid12,

stores\_store\_id );

CREATE TABLE transactions (

patient\_patid VARCHAR2(100) NOT NULL,

stores\_store\_id VARCHAR2(100) NOT NULL,

quantity VARCHAR2(200),

purchasedate DATE,

total INTEGER,

patid VARCHAR2(100) NOT NULL

);

CREATE UNIQUE INDEX transactions\_\_idx ON

transactions (

patient\_patid

ASC,

stores\_store\_id

ASC );

ALTER TABLE transactions ADD CONSTRAINT transactions\_pk PRIMARY KEY ( patient\_patid,

stores\_store\_id );

CREATE TABLE treatment (

doctor\_docid VARCHAR2(100) NOT NULL,

hospital\_hosid VARCHAR2(100) NOT NULL,

treatid VARCHAR2(100) NOT NULL,

treatdate DATE,

patient\_patid12 VARCHAR2(100) NOT NULL,

patient\_patid1 VARCHAR2(200) NOT NULL

);

CREATE UNIQUE INDEX treatment\_\_idx ON

treatment (

treatid

ASC );

ALTER TABLE treatment ADD CONSTRAINT treatment\_pk PRIMARY KEY ( doctor\_docid,

hospital\_hosid );

ALTER TABLE contract

ADD CONSTRAINT contract\_hospital\_fk FOREIGN KEY ( hospital\_hosid )

REFERENCES hospital ( hosid );

ALTER TABLE contract

ADD CONSTRAINT contract\_stores\_fk FOREIGN KEY ( stores\_store\_id )

REFERENCES stores ( store\_id );

ALTER TABLE retail

ADD CONSTRAINT retail\_medicine\_fk FOREIGN KEY ( medicine\_med\_id )

REFERENCES medicine ( med\_id )

ON DELETE CASCADE;

ALTER TABLE retail

ADD CONSTRAINT retail\_stores\_fk FOREIGN KEY ( stores\_store\_id )

REFERENCES stores ( store\_id )

ON DELETE CASCADE;

ALTER TABLE supplies

ADD CONSTRAINT supplies\_dealer\_fk FOREIGN KEY ( dealer\_dealid12 )

REFERENCES dealer ( dealid );

ALTER TABLE supplies

ADD CONSTRAINT supplies\_stores\_fk FOREIGN KEY ( stores\_store\_id )

REFERENCES stores ( store\_id );

ALTER TABLE transactions

ADD CONSTRAINT transactions\_patient\_fk FOREIGN KEY ( patient\_patid )

REFERENCES patient ( patid );

ALTER TABLE transactions

ADD CONSTRAINT transactions\_stores\_fk FOREIGN KEY ( stores\_store\_id )

REFERENCES stores ( store\_id );

ALTER TABLE treatment

ADD CONSTRAINT treatment\_doctor\_fk FOREIGN KEY ( doctor\_docid )

REFERENCES doctor ( docid );

ALTER TABLE treatment

ADD CONSTRAINT treatment\_hospital\_fk FOREIGN KEY ( hospital\_hosid )

REFERENCES hospital ( hosid );

ALTER TABLE treatment

ADD CONSTRAINT treatment\_patient\_fk FOREIGN KEY ( patient\_patid12 )

REFERENCES patient ( patid );

# Tables:

**Graphical user interface, text

Description automatically generatedTreatment:**

**Graphical user interface, text

Description automatically generatedTransaction:**

**Graphical user interface

Description automatically generatedSupplies:**

**Stores:**

Graphical user interface, text, application

Description automatically generated

**Retail:**

Graphical user interface, table

Description automatically generated

**Patient:**

Graphical user interface, text

Description automatically generated

**Medicine:**

Graphical user interface

Description automatically generated

**Hospital:**

Graphical user interface, text

Description automatically generated

**Doctor:**

Graphical user interface

Description automatically generated

**Dealer:**

Graphical user interface, text

Description automatically generated

**Contract:**

Graphical user interface, text

Description automatically generated

# SQL QUERIES:

**Concatenation with literal character string:**

Graphical user interface, text, application, email

Description automatically generated

**Using Describe keyword query:**

Graphical user interface, text, application

Description automatically generated

**Range condition using the between operator with where clause:**

Graphical user interface, text, application

Description automatically generated

**Using Like operator to find name and address of the dealer whose third letter of the name is m:**

Graphical user interface, text, application, email

Description automatically generated

**Using Aggregate function for minimum, maximum, average and sum of the total in transaction with condition on quantity is between 5 and 13:**

Graphical user interface, text, application

Description automatically generated

**In this statement, we have to use the Natural join to display patient id, store id, name of the patient and address of the patient from the two tables transaction and patient in which patient id is pat1:**

Graphical user interface, text, application, email

Description automatically generated

**Use of natural join between 3 tables:**

SELECT

"A1"."DOCTOR\_DOCID" "DOCTOR\_DOCID",

"A1"."HOSPITAL\_HOSID" "HOSPITAL\_HOSID",

"A1"."TREATID" "TREATID",

"A1"."TREATDATE" "TREATDATE",

"A1"."PATIENT\_PATID1" "PATIENT\_PATID1",

"A1"."PATID" "PATID",

"A1"."NAME" "NAME",

"A1"."ADDRESS" "ADDRESS",

"A1"."PHONENO" "PHONENO",

"A1"."DOCID" "DOCID",

"A1"."DOCNAME" "DOCNAME"

FROM

(

SELECT

"A3"."DOCTOR\_DOCID" "DOCTOR\_DOCID",

"A3"."HOSPITAL\_HOSID" "HOSPITAL\_HOSID",

"A3"."TREATID" "TREATID",

"A3"."TREATDATE" "TREATDATE",

"A3"."PATIENT\_PATID1" "PATIENT\_PATID1",

"A3"."PATID" "PATID",

"A3"."NAME" "NAME",

"A3"."ADDRESS" "ADDRESS",

"A3"."PHONENO" "PHONENO",

"A2"."DOCID" "DOCID",

"A2"."DOCNAME" "DOCNAME"

FROM

(

SELECT

"A5"."DOCTOR\_DOCID" "DOCTOR\_DOCID",

"A5"."HOSPITAL\_HOSID" "HOSPITAL\_HOSID",

"A5"."TREATID" "TREATID",

"A5"."TREATDATE" "TREATDATE",

"A5"."PATIENT\_PATID1" "PATIENT\_PATID1",

"A4"."PATID" "PATID",

"A4"."NAME" "NAME",

"A4"."ADDRESS" "ADDRESS",

"A4"."PHONENO" "PHONENO"

FROM

"GROUPPROJECT"."TREATMENT" "A5",

"GROUPPROJECT"."PATIENT" "A4"

) "A3",

"GROUPPROJECT"."DOCTOR" "A2"

) "A1"

WHERE

"A1"."PATID" = "A1"."PATIENT\_PATID1"

Graphical user interface, text, application

Description automatically generated AND "A1"."DOCTOR\_DOCID" = "A1"."DOCID";

**Retrieving data using the SQL SELECT statement:**

1. **Using column aliases:**

SELECT

"A1"."NAME" "HOSPITALNAME",

"A1"."ADDRESS" "HOSPITALADDRESS",

"A1"."PHONENO" "HOSPITALCONTACT"

FROM

"GROUPPROJECT"."HOSPITAL" "A1";

**Output:**

**Graphical user interface, text, table

Description automatically generated**

1. **Using concatenation operator:**

SELECT

"A2".NAME||"A2".COST\_PER\_TAB AS NameAndCosts

FROM "GROUPPROJECT"."MEDICINE" "A2";

**Output:**

A picture containing graphical user interface

Description automatically generated

1. **Using literal character string:**

SELECT

'The phone number of patient '

|| "A1"."NAME"

|| ' is '

|| "A1"."PHONENO" "PATIENTCONTACT"

FROM

"GROUPPROJECT"."PATIENT" "A1";

**Output:**

**Text, table

Description automatically generated**

1. **Use of distinct keyword:**

**Actual Data in store table:**

Graphical user interface, table

Description automatically generated

As in this table “Umair Pharmacy” is repeated two times as in S1 and S9, if we only select “name” and “manager” So the distinct keyword remove that duplicate from the table and output the result.

So the query is:

SELECT DISTINCT

"A1"."NAME" "NAME",

"A1"."MANAGER" "MANAGER"

FROM

"GROUPPROJECT"."STORES" "A1";

**Output:**

**Table

Description automatically generated**

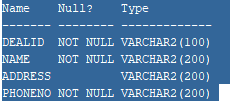
1. **Use of DESCRIBE keyword:**

To display the structure of dealer table we use the describe keyword.

Query:

DESCRIBE dealer;

**Output:**

****

**Restricting and sorting data:**

1. **Using the WHERE clause and comparison operator:**

Display the medicine whose cost\_per\_tab is less than or equal to 30.

SELECT

"A1"."NAME" "NAME"

FROM

"GROUPPROJECT"."MEDICINE" "A1"

WHERE

"A1"."COST\_PER\_TAB" <= 30;

**Output:**

**Graphical user interface

Description automatically generated**

1. **Using the WHERE clause with character String:**

* Display the pharmacy name whose manager is “UMAIR”.

**Query:**

SELECT

"A1"."NAME" "NAME",

"A1"."ADDRESS" "ADDRESS",

"A1"."CONTACT\_NO" "CONTACT\_NO"

FROM

"GROUPPROJECT"."STORES" "A1"

WHERE

"A1"."MANAGER" = 'Umair';

**Output:**

Table

Description automatically generated

1. **Range condition using the BETWEEN OPERATOR:**

SELECT

"A1"."NAME" "NAME",

"A1"."MFG\_DATE" "MFG\_DATE",

"A1"."EXP\_DATE" "EXP\_DATE",

"A1"."COST\_PER\_TAB" "COST\_PER\_TAB"

FROM

"GROUPPROJECT"."MEDICINE" "A1"

WHERE

"A1"."COST\_PER\_TAB" BETWEEN 30 AND 90;

**OUTPUT:**

**Table

Description automatically generated with medium confidence**

1. **Combining conditions using the logical operator:**

**Use of AND operator:**

Same query as 3 but with logical AND operator:

Query:

SELECT

"A1"."NAME" "NAME",

"A1"."MFG\_DATE" "MFG\_DATE",

"A1"."EXP\_DATE" "EXP\_DATE",

"A1"."COST\_PER\_TAB" "COST\_PER\_TAB"

FROM

"GROUPPROJECT"."MEDICINE" "A1"

WHERE

"A1"."COST\_PER\_TAB" >= 30

AND "A1"."COST\_PER\_TAB" <= 90;

**OUTPUT:**

Table

Description automatically generated with medium confidence

1. **Sorting rows using the order by clause:**

Sorting “Store” table using the “Contact-no” and then “StoreId”.

**Query:**

SELECT

"A1"."NAME" "NAME",

"A1"."MANAGER" "MANAGER",

"A1"."ADDRESS" "ADDRESS",

"A1"."CONTACT\_NO" "CONTACT\_NO"

FROM

"GROUPPROJECT"."STORES" "A1"

ORDER BY

"A1"."CONTACT\_NO",

"A1"."STORE\_ID";

**OUTPUT:**

**Table

Description automatically generated**

**Using Single row Function to customize Output:**

1. **Use of date Functions:**

Display the number of weeks we can use the tablet:

**Query:**

SELECT

"A1"."NAME" "NAME",

( "A1"."EXP\_DATE" - "A1"."MFG\_DATE" ) / 7 "WEEKFORUSE"

FROM

"GROUPPROJECT"."MEDICINE" "A1";

**OUTPUT:**

Text

Description automatically generated

**Reported aggregated data using the group function:**

1. **Using aggregate to find min, max, sum, avg and counts of per tab.**

**Query:**

SELECT

COUNT(\*) "COUNTIDS",

AVG("A1"."COST\_PER\_TAB") "AVERAGECOST",

MIN("A1"."COST\_PER\_TAB") "MINIMUMCOST",

MAX("A1"."COST\_PER\_TAB") "MAXIMUMCOST",

SUM("A1"."COST\_PER\_TAB") "SUMOFCOSTS"

FROM

"GROUPPROJECT"."MEDICINE" "A1"

WHERE

"A1"."MED\_ID" >= 'M4'

AND "A1"."MED\_ID" <= 'M9';

**Output:**



1. **Creating group of data:**

Select the name and composition of the “Medicine” table and sum the “cost per tab” of the medicine table whose formula is same and then use the order by clause.

**Query:**

SELECT

"A1"."COMPOSITION" "COMPOSITION",

SUM("A1"."COST\_PER\_TAB") "SUM(COST\_PER\_TAB)"

FROM

"GROUPPROJECT"."MEDICINE" "A1"

GROUP BY

"A1"."COMPOSITION"

ORDER BY

"A1"."COMPOSITION";

**OUTPUT:**

Table

Description automatically generated

**Displaying data from multiple tables using joins:**

1. **Use of join:**

SELECT patient.patid,transactions.patient\_patid,patient.name,transactions.quantity

FROM transactions JOIN patient

on transactions.patient\_patid = patient.patid

ORDER BY patient.patid;

**Output:**

**Table

Description automatically generated**

1. **Use of natural join:**

**Query:**

SELECT patid,patient.name,patient.address,patient.phoneno,transactions.purchasedate,transactions.total

FROM transactions NATURAL JOIN patient;

**Output:**

A picture containing text, window

Description automatically generated

1. **Creating joins with the using clause:**

SELECT patid,patient.name,patient.address,patient.phoneno,transactions.purchasedate,transactions.total

FROM transactions JOIN patient USING (patid);

**Output:**

**Table

Description automatically generated**

1. **Creating join with ON clause:**

SELECT st.store\_id,st.name,st.manager,st.contact\_no

FROM stores st JOIN supplies su

ON st.store\_id = su.store\_id

ORDER BY st.store\_id;

**OUTPUT:**

Table

Description automatically generated

**Using subqueries to solve queries:**

1. **Use of single row query:**

**Query:**

SELECT quantity,total

FROM transactions

WHERE transactions.patid = ( SELECT patient.patid

FROM patient

WHERE patient.patid = 'pat1' );

**Output:**

Table

Description automatically generated

1. **Use of subquery with AND operator**

**Query:**

SELECT patient.patid,name,address,phoneno

FROM patient

WHERE patient.patid = ( SELECT patid

FROM transactions

WHERE transactions.stores\_store\_id = 'S2' AND transactions.patid = 'pat1' );

**Output:**



1. **Use of IN operator with subquery:**

**Query:**

SELECT patient.patid,name,address,phoneno

FROM patient

WHERE patient.patid IN

( SELECT transactions.patid

FROM transactions

WHERE transactions.patid = 'pat1' );

**OUTPUT:**

****

**Using the Set Operator:**

1. **Union Operator:**

**Query:**

( SELECT

"A3"."PATID" "PATID"

FROM

"GROUPPROJECT"."PATIENT" "A3"

)

UNION

( SELECT

"A2"."PATID" "PATID"

FROM

"GROUPPROJECT"."TRANSACTIONS" "A2"

);

**Output:**

**Table

Description automatically generated**

1. **Using the minus operator with subquery:**

**Query:**

SELECT

"A1"."PATID" "PATID",

"A1"."NAME" "NAME",

"A1"."ADDRESS" "ADDRESS"

FROM

"GROUPPROJECT"."PATIENT" "A1"

WHERE

"A1"."PATID" = ANY (

( SELECT

"A4"."PATID" "PATID"

FROM

"GROUPPROJECT"."PATIENT" "A4"

)

MINUS

( SELECT

"A3"."PATID" "PATID"

FROM

"GROUPPROJECT"."TRANSACTIONS" "A3"

)

);

**OUTPUT:**

**Table

Description automatically generated with medium confidence**

**Manipulating data using DML statement:**

1. **Inserting new rows:**

Insert into dealer table:

INSERT INTO dealer (dealid,name,address,phoneno)

VALUES ('D13','kashif','gujjar khan','0301-2222223');

**Output:**

****

Table:

Table

Description automatically generated

1. **Update query:**

UPDATE dealer

SET name = 'Hussain'

WHERE dealid = 'D13';

**Output:**



Table:

Table

Description automatically generated

1. **Row Delete query**

DELETE FROM dealer

WHERE dealid = 'D13';

**Output:**



Table now:

Table

Description automatically generated

# Mongo DB:

**Patient Record:**

[{

"\_id": {

"$oid": "61c55b7a1455918efc35f7ae"

},

"patientname": "Hamza",

"patientaddress": "Karachi",

"patientcontact": "0302-2222222",

"Doctor": {

"doctorname": "Tahir"

},

"HospitalInfo": {

"hospitalname": "Center Hospital",

"hospitaladdress": "Rawalpindi",

"hospitalcontact": "0303-1111112",

"treatmentdate": {

"$date": "2021-11-30T19:00:00Z"

}

},

"Pharmacy": {

"pharmacyname": "Umair Pharmacy",

"pharmacyowner": "Umair",

"pharmacyaddress": "Rawalpindi",

"pharmacycontacy": "0333-1111111",

"purchasedate": "07-DEC-21",

"quantity": "12",

"totalbill": 1345

}

},{

"\_id": {

"$oid": "61c55c341455918efc35f7af"

},

"patientname": "Arham",

"patientaddress": "Rawalpindi",

"patientcontact": "0302-2222220",

"Doctor": {

"doctorname": "Tahir"

},

"Hospital": {

"hospitalname": "CMH",

"hospitaladdress": "Rawalpindi",

"hospitalcontact": "0303-1111110",

"treatmentdate": "01-DEC-21"

},

"Pharmacy": {

"pharmacyname": "Arshad Pharmacy",

"pharmacyowner": "Arshad",

"pharmacyaddress": "Islamabad",

"pharmacycontacy": "0333-1111112",

"purchasedate": {

"$date": "2021-12-06T19:00:00Z"

},

"quantity": "4",

"totalbill": 1455

}

},{

"\_id": {

"$oid": "61c55ce21455918efc35f7b0"

},

"patientname": "Zamin",

"patientaddress": "Rawalpindi",

"patientcontact": "0302-2222221",

"Doctor": {

"doctorname": "Tahir"

},

"Hospital": {

"hospitalname": "MH",

"hospitaladdress": "Rawalpindi",

"hospitalcontact": "0303-1111111",

"treatmentdate": "01-DEC-21"

},

"Pharmacy": {

"pharmacyname": "Arshad Pharmacy",

"pharmacyowner": "Arshad",

"pharmacyaddress": "Islamabad",

"pharmacycontacy": "0333-1111112",

"purchasedate": {

"$date": "2021-12-06T19:00:00Z"

},

"quantity": "8",

"totalbill": 1300

}

},{

"\_id": {

"$oid": "61c55d581455918efc35f7b1"

},

"patientname": "Hamza",

"patientaddress": "Karachi",

"patientcontact": "0302-2222222",

"Doctor": {

"doctorname": "Tahir"

},

"Hospital": {

"hospitalname": "Center Hospital",

"hospitaladdress": "Rawalpindi",

"hospitalcontact": "0303-1111112",

"treatmentdate": "01-DEC-21"

},

"Pharmacy": {

"pharmacyname": "Arshad Pharmacy",

"pharmacyowner": "Arshad",

"pharmacyaddress": "Islamabad",

"pharmacycontacy": "0333-1111112",

"purchasedate": {

"$date": "2021-12-06T19:00:00Z"

},

"quantity": "13",

"totalbill": 1200

}

},{

"\_id": {

"$oid": "61c55dc31455918efc35f7b2"

},

"patientname": "Arham",

"patientaddress": "Rawalpindi",

"patientcontact": "0302-2222220",

"Doctor": {

"doctorname": "Tahir"

},

"Hospital": {

"hospitalname": "CMH",

"hospitaladdress": "Rawalpindi",

"hospitalcontact": "0303-1111110",

"treatmentdate": "01-DEC-21"

},

"Pharmacy": {

"pharmacyname": "Zeeshan Pharmacy",

"pharmacyowner": "Zeeshan",

"pharmacyaddress": "Rawalpindi",

"pharmacycontacy": "0333-1111113",

"purchasedate": {

"$date": "2021-12-06T19:00:00Z"

},

"quantity": "5",

"totalbill": 1400

}

},{

"\_id": {

"$oid": "61c55e1c1455918efc35f7b3"

},

"patientname": "Zamin",

"patientaddress": "Rawalpindi",

"patientcontact": "0302-2222221",

"Doctor": {

"doctorname": "Tahir"

},

"Hospital": {

"hospitalname": "MH",

"hospitaladdress": "Rawalpindi",

"hospitalcontact": "0303-1111111",

"treatmentdate": "01-DEC-21"

},

"Pharmacy": {

"pharmacyname": "Zeeshan Pharmacy",

"pharmacyowner": "Zeeshan",

"pharmacyaddress": "Rawalpindi",

"pharmacycontacy": "0333-1111113",

"purchasedate": {

"$date": "2021-12-06T19:00:00Z"

},

"quantity": "9",

"totalbill": 1200

}

},{

"\_id": {

"$oid": "61c55e721455918efc35f7b4"

},

"patientname": "Irfan",

"patientaddress": "Karachi",

"patientcontact": "0302-2222223",

"Doctor": {

"doctorname": "Tahir"

},

"Hopital": {

"hospitalname": "Benazir Hospital",

"hospitaladdress": "Rawalpindi",

"hospitalcontact": "0303-1111113",

"treatmentdate": "01-DEC-21"

},

"Pharmacy": {

"pharmacyname": "Zeeshan Pharmacy",

"pharmacyowner": "Zeeshan",

"pharmacyaddress": "Rawalpindi",

"pharmacycontacy": "0333-1111113",

"purchasedate": {

"$date": "2021-12-06T19:00:00Z"

},

"quantity": "14",

"totalbill": 1600

}

},{

"\_id": {

"$oid": "61c55eb91455918efc35f7b5"

},

"patientname": "Arham",

"patientaddress": "Rawalpindi",

"patientcontact": "0302-2222220",

"Doctor": {

"doctorname": "Tahir"

},

"Hospital": {

"hospitalname": "CMH",

"hospitaladdress": "Rawalpindi",

"hospitalcontact": "0303-1111110",

"treatmentdate": "01-DEC-21"

},

"Pharmacy": {

"pharmacyname": "Mujtaba Pharmacy",

"pharmacyowner": "Mujtaba",

"pharmacyaddress": "Islamabad",

"pharmacycontacy": "0333-1111110",

"purchasedate": {

"$date": "2021-12-06T19:00:00Z"

},

"quantity": "6",

"totalbill": 560

}

},{

"\_id": {

"$oid": "61c55eeb1455918efc35f7b6"

},

"patientname": "Zamin",

"patientaddress": "Rawalpindi",

"patientcontact": "0302-2222221",

"Doctor": {

"doctorname": "Tahir"

},

"Hospital": {

"hospitalname": "MH",

"hospitaladdress": "Rawalpindi",

"hospitalcontact": "0303-1111111",

"treatmentdate": "01-DEC-21"

},

"Pharmacy": {

"pharmacyname": "Mujtaba Pharmacy",

"pharmacyowner": "Mujtaba",

"pharmacyaddress": "Islamabad",

"pharmacycontacy": "0333-1111110",

"purchasedate": {

"$date": "2002-12-06T19:00:00Z"

},

"quantity": "10",

"totalbill": 300

}

},{

"\_id": {

"$oid": "61c55f3b1455918efc35f7b7"

},

"patientname": "Zamin",

"patientaddress": "Rawalpindi",

"patientcontact": "0302-2222221",

"Doctor": {

"doctorname": "Tahir"

},

"Hospital": {

"hospitalname": "MH",

"hospitaladdress": "Rawalpindi",

"hospitalcontact": "0303-1111111",

"treatmentdate": "01-DEC-21"

},

"Pharmacy": {

"pharmacyname": "Shaharyar Pharmacy",

"pharmacyowner": "Shaharyar",

"pharmacyaddress": "Karachi",

"pharmacycontacy": "0333-1111114",

"purchasedate": {

"$date": "2021-12-06T19:00:00Z"

},

"quantity": "11",

"totalbill": 1500

}

},{

"\_id": {

"$oid": "61c55f721455918efc35f7b8"

},

"patientname": "Arham",

"patientaddress": "Rawalpindi",

"patientcontact": "0302-2222220",

"doctor": {

"doctorname": "Tahir"

},

"hospital": {

"hospitalname": "CMH",

"hospitaladdress": "Rawalpindi",

"hospitalcontact": "0303-1111110",

"treatmentdate": "01-DEC-21"

},

"pharmacy": {

"pharmacyname": "Umair Pharmacy",

"pharmacyowner": "Umair",

"pharmacyaddress": "Rawalpindi",

"pharmacycontacy": "0333-1111111",

"purchasedate": {

"$date": "2021-12-06T19:00:00Z"

},

"quantity": "3",

"totalbill": 1300

}

},{

"\_id": {

"$oid": "61c55fe01455918efc35f7b9"

},

"patientname": "Arham",

"patientaddress": "Rawalpindi",

"patientcontact": "0302-2222220",

"Doctor": {

"doctorname": "Tahir"

},

"Hospital": {

"hospitalname": "CMH",

"hospitaladdress": "Rawalpindi",

"hospitalcontact": "0303-1111110",

"treatmentdate": "01-DEC-21"

},

"Pharmacy": {

"pharmacyname": "Umair Pharmacy",

"pharmacyowner": "Umair",

"pharmacyaddress": "Rawalpindi",

"pharmacycontacy": "0333-1111111",

"purchasedate": {

"$date": "2021-12-06T19:00:00Z"

},

"quantity": "3",

"totalbill": 1300

}

}]

**Store Record:**

[{

"\_id": {

"$oid": "61bc2faa642c1a85790f86d2"

},

"storename": "Umair Pharmacy",

"storemanager": "Umair",

"storecontact": "0333-1111111",

"storeaddress": "Rawalpindi",

"medicinename": "Panadol",

"mfg\_date": "01-DEC-20",

"exp\_date": "01-DEC-23",

"composition": "Formula1",

"cost\_per\_tab": 30,

"quantity": 4,

"dealername": "Bahadur",

"dealraddress": "Gujranwala",

"dealercontact": "0301-1111113"

},{

"\_id": {

"$oid": "61bc2fc2642c1a85790f86d3"

},

"storename": "Arshad Pharmacy",

"storemanager": "Arshad",

"storecontact": "0333-1111112",

"storeaddress": "Islamabad",

"medicinename": "Panadol",

"mfg\_date": "01-DEC-20",

"exp\_date": "01-DEC-23",

"composition": "Formula1",

"cost\_per\_tab": 30,

"quantity": 12,

"dealername": "Usman",

"dealraddress": "Rawalpindi",

"dealercontact": "0301-1111111"

},{

"\_id": {

"$oid": "61bc2fd1642c1a85790f86d4"

},

"storename": "Zeeshan Pharmacy",

"storemanager": "Zeeshan",

"storecontact": "0333-1111113",

"storeaddress": "Rawalpindi",

"medicinename": "Panadol",

"mfg\_date": "01-DEC-20",

"exp\_date": "01-DEC-23",

"composition": "Formula1",

"cost\_per\_tab": 30,

"quantity": 5,

"dealername": "Khurram",

"dealraddress": "Rawalpindi",

"dealercontact": "0301-1111110"

},{

"\_id": {

"$oid": "61bc2fe0642c1a85790f86d5"

},

"storename": "Umair Pharmacy",

"storemanager": "Umair",

"storecontact": "0333-1111111",

"storeaddress": "Rawalpindi",

"medicinename": "Panadol Extra",

"mfg\_date": "01-DEC-20",

"exp\_date": "01-DEC-23",

"composition": "Formula1",

"cost\_per\_tab": 40,

"quantity": 10,

"dealername": "Bahadur",

"dealraddress": "Gujranwala",

"dealercontact": "0301-1111113"

},{

"\_id": {

"$oid": "61bc2fee642c1a85790f86d6"

},

"storename": "Arshad Pharmacy",

"storemanager": "Arshad",

"storecontact": "0333-1111112",

"storeaddress": "Islamabad",

"medicinename": "Panadol Extra",

"mfg\_date": "01-DEC-20",

"exp\_date": "01-DEC-23",

"composition": "Formula1",

"cost\_per\_tab": 40,

"quantity": 11,

"dealername": "Usman",

"dealraddress": "Rawalpindi",

"dealercontact": "0301-1111111"

},{

"\_id": {

"$oid": "61bc2ffb642c1a85790f86d7"

},

"storename": "Zeeshan Pharmacy",

"storemanager": "Zeeshan",

"storecontact": "0333-1111113",

"storeaddress": "Rawalpindi",

"medicinename": "Panadol Extra",

"mfg\_date": "01-DEC-20",

"exp\_date": "01-DEC-23",

"composition": "Formula1",

"cost\_per\_tab": 40,

"quantity": 9,

"dealername": "Khurram",

"dealraddress": "Rawalpindi",

"dealercontact": "0301-1111110"

},{

"\_id": {

"$oid": "61bc300d642c1a85790f86d8"

},

"storename": "Mujtaba Pharmacy",

"storemanager": "Mujtaba",

"storecontact": "0333-1111110",

"storeaddress": "Islamabad",

"medicinename": "Panadol Extra",

"mfg\_date": "01-DEC-20",

"exp\_date": "01-DEC-23",

"composition": "Formula1",

"cost\_per\_tab": 40,

"quantity": 8,

"dealername": "Sardar",

"dealraddress": "Sialkot",

"dealercontact": "0301-1111112"

},{

"\_id": {

"$oid": "61bc301b642c1a85790f86d9"

},

"storename": "Bukhari Pharmacy",

"storemanager": "Bukhari",

"storecontact": "0333-1111116",

"storeaddress": "Lahore",

"medicinename": "Disprin",

"mfg\_date": "01-DEC-20",

"exp\_date": "01-DEC-23",

"composition": "Formula3",

"cost\_per\_tab": 30,

"quantity": 12,

"dealername": "Kamran",

"dealraddress": "Gujranwala",

"dealercontact": "0301-1111116"

},{

"\_id": {

"$oid": "61bc302f642c1a85790f86da"

},

"storename": "Umair Pharmacy",

"storemanager": "Umair",

"storecontact": "0333-1111111",

"storeaddress": "Rawalpindi",

"medicinename": "Disprin",

"mfg\_date": "01-DEC-20",

"exp\_date": "01-DEC-23",

"composition": "Formula3",

"cost\_per\_tab": 30,

"quantity": 3,

"dealername": "Shahzad",

"dealraddress": "Gujranwala",

"dealercontact": "0301-1111117"

},{

"\_id": {

"$oid": "61bc3035642c1a85790f86db"

},

"storename": "Tariq Pharmacy",

"storemanager": "Tariq",

"storecontact": "0333-1111121",

"storeaddress": "Lahore",

"medicinename": "Disprin",

"mfg\_date": "01-DEC-20",

"exp\_date": "01-DEC-23",

"composition": "Formula3",

"cost\_per\_tab": 30,

"quantity": 5,

"dealername": "Danish",

"dealraddress": "Kashmir",

"dealercontact": "0301-1111119"

},{

"\_id": {

"$oid": "61bc3044642c1a85790f86dc"

},

"storename": "Mujtaba Pharmacy",

"storemanager": "Mujtaba",

"storecontact": "0333-1111110",

"storeaddress": "Islamabad",

"medicinename": "Panadol Extra",

"mfg\_date": "01-DEC-20",

"exp\_date": "01-DEC-23",

"composition": "Formula1",

"cost\_per\_tab": 30,

"quantity": 1,

"dealername": "Sardar",

"dealraddress": "Sialkot",

"dealercontact": "0301-1111112"

},{

"\_id": {

"$oid": "61bc3055642c1a85790f86dd"

},

"storename": "Umair Pharmacy",

"storemanager": "Umair",

"storecontact": "0333-1111111",

"storeaddress": "Rawalpindi",

"medicinename": "Panadol Extra",

"mfg\_date": "01-DEC-20",

"exp\_date": "01-DEC-23",

"composition": "Formula1",

"cost\_per\_tab": 30,

"quantity": 34,

"dealername": "Bahadur",

"dealraddress": "Gujranwala",

"dealercontact": "0301-1111113"

}]

# Mongo Db Queries:

**Query to insert document and document within another document in the collection:**

db.PatientRecord.insertOne({'patientname': 'Ahmad','patientaddress': 'Islamabad','patientcontact': '0302-2222231','Doctor':{'doctorname': 'Hassan'},'Hospital':{'hospitalname':'Center','hospitaladdress':'Karachi','treatmentdate':'28-DEC-2021'},'Pharmacy':{'pharmacyname':'sharoze pharmacy','pharmacyowner':'sharoze','pharmacyaddress':'Karachi'}})

**Query to show all rows in a collection in a prettified manner:**

Graphical user interface, text

Description automatically generated

**Query to find only one row matching the object:**

Graphical user interface, text

Description automatically generated

**Find the pharmacy information of the patient whose name is Hamza:**

Text

Description automatically generated

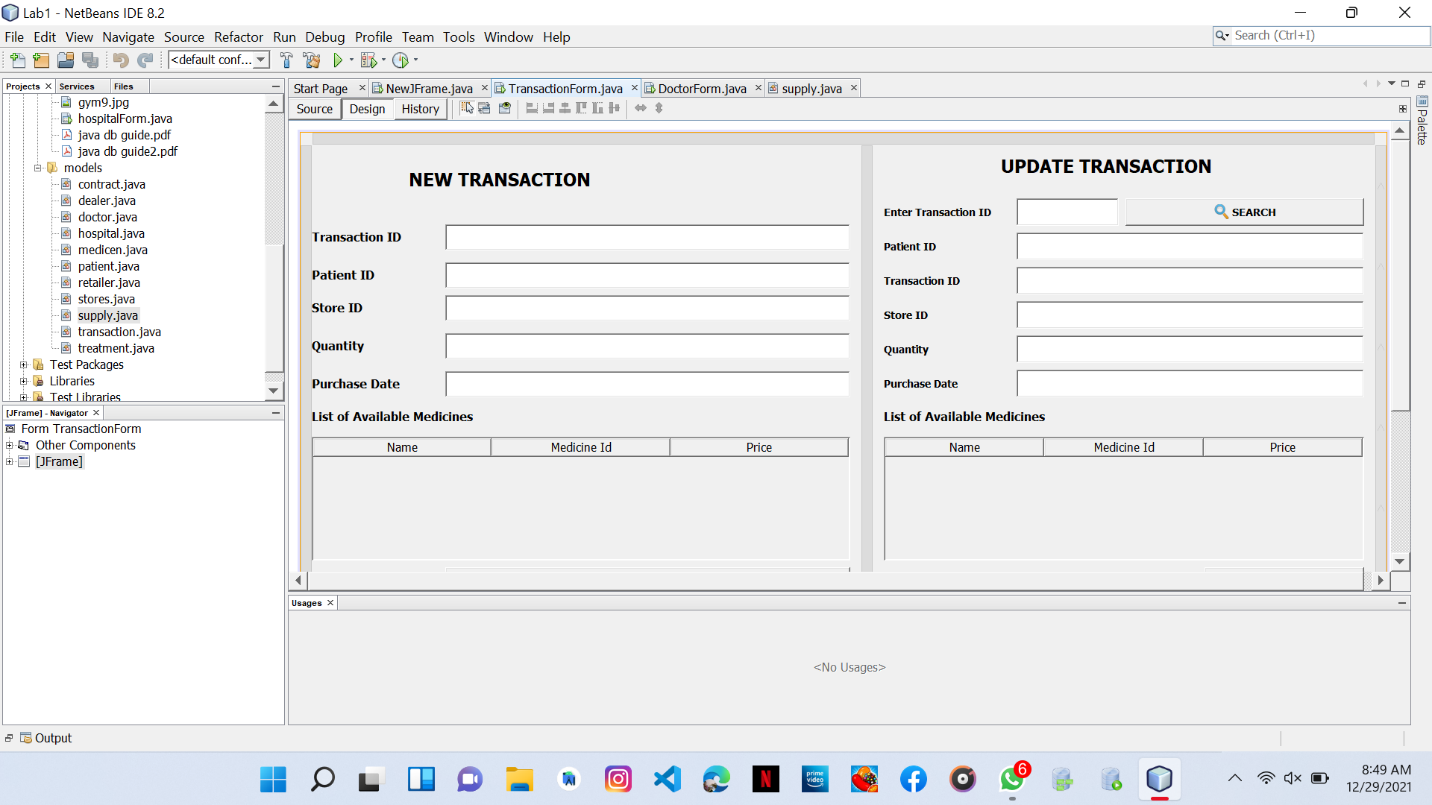
# Mockup:

**Main Menu:**

Graphical user interface, text, application

Description automatically generated

**Transaction:**



**Update Record for Doctor:**

Graphical user interface, text, application

Description automatically generated

**Record for Doctors:**

Graphical user interface, text, application

Description automatically generated

**Update Treatment Form:**

Graphical user interface, application

Description automatically generated

**Treatment Records:**

A screenshot of a computer

Description automatically generated

**Patient Records:**

Graphical user interface, text

Description automatically generated

**Add Patient:**

Graphical user interface, text, application

Description automatically generated

**Update Retail:**

Graphical user interface, text, application

Description automatically generated

**Update Supply Record:**

Graphical user interface, text, application

Description automatically generated

**Update Transaction:**

Graphical user interface, text, application

Description automatically generated

**New Transaction:**

Graphical user interface, text, application

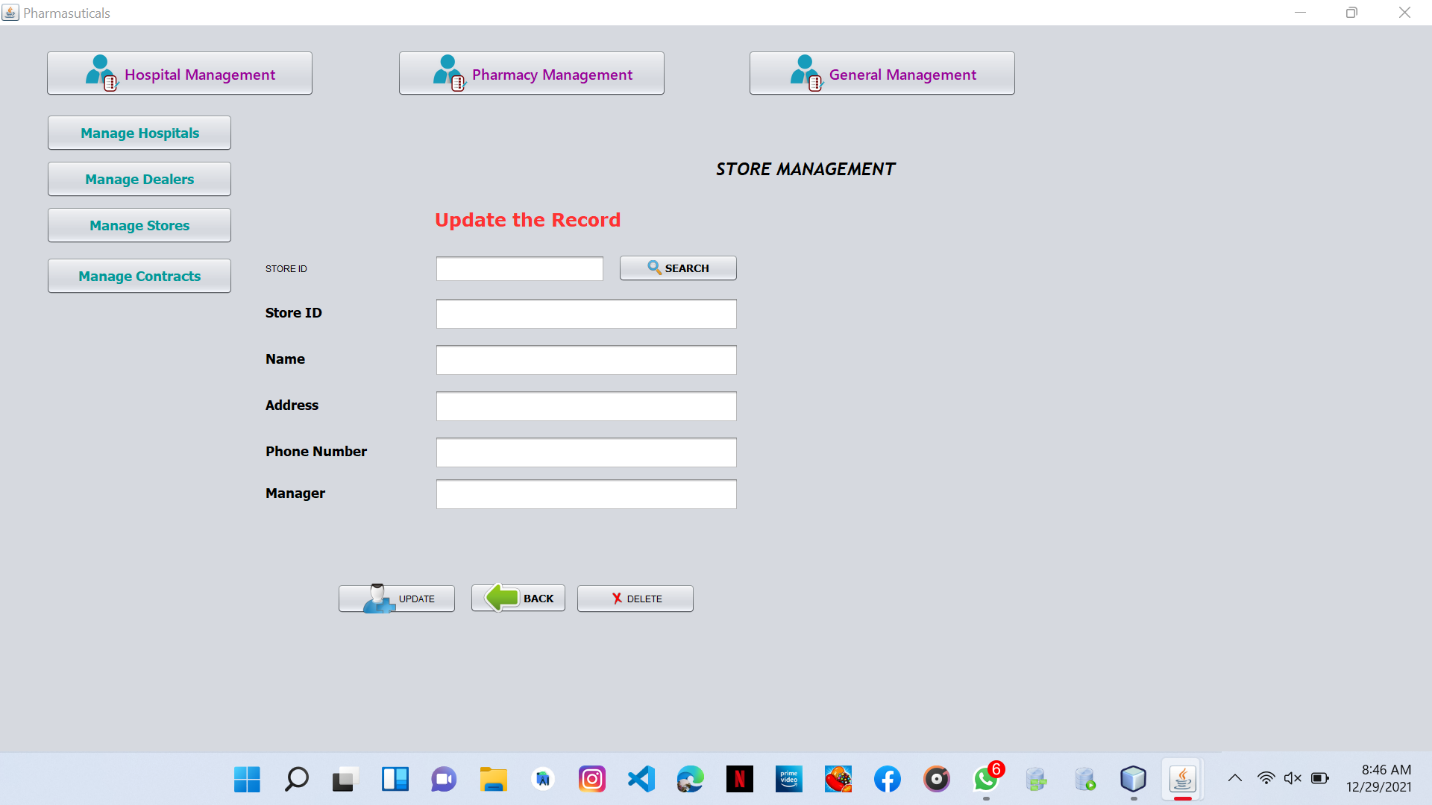
Description automatically generated

**Update Records For Contract Management:**

Graphical user interface, text

Description automatically generated

**Update Record for Store Management:**



**Update Record for Dealer Management:**

Graphical user interface, text

Description automatically generated

**Add New Hospital:**

Graphical user interface, text, application

Description automatically generated