

Department of CSE

Database Management Systems

# Project Report

## Pharmaceutical - Retail Management System

- Ishaan Kumar
- Roll-22CSB0A02
- Sec-A
  
- Vivek Kanaujiya
- Roll-22CSB0B29
- Sec-B

## PROJECT SUMMARY

The title of this project is Pharmaceutical-Retail Management system where pharma stores have contracts with hospitals. The retail part of the pharma store that the patient buying medicine from the stores are focused. The schema consists of 10 relations (5 strong entities + 1 weak entity + 4 relationships). The Data Model and the Relational Schema have been discussed below. The schema is normalized until the third normal form (3NF) and the process that led to normalizing until 3NF is given in the upcoming sections.

DDL commands have been put in the DDL sections, which were used to create the schema of our database.

Thus this database system is capable of performing most of the actions related to a Pharmaceutical-Retail system, where emphasis has been laid on both the Pharma and Retail sectors. Limitations and Future work have been discussed thoroughly in the conclusion section.

## ➤ Table Of Contents

Section	Page Range
<a href="#">Introduction</a>	3 - 4
<a href="#">Data Model</a>	5 - 6
<a href="#">Normalization</a>	7 - 8
<a href="#">DDL</a>	9 - 11
<a href="#">Insertion</a>	12 - 14
<a href="#">SQL Queries</a>	15 - 16
<a href="#">Conclusion</a>	17

## ➤ 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.

### 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.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.

#### 3.DOCTOR

Contains information like doctor id(*doc\_id*), and doctor name(*doc\_name*).

#### 4.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.

#### 5.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.

#### 6.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*) and a total on a particular . This entire process, between the PATIENT and the STORES, has been modeled into this single relation.

### 7.RETAIL

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

### 8.CONTRACT

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

### 9. STORES

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

### 10. QUANT

It keeps a track of all the medicines in all the stores along with its quantity. This will be useful in notifying when to replenish the medicine stocks.

## ➤ Data Model

The Entity-Relationship Diagram (ERD) for the Pharmaceutical-Retail Management System is shown in Fig 1.

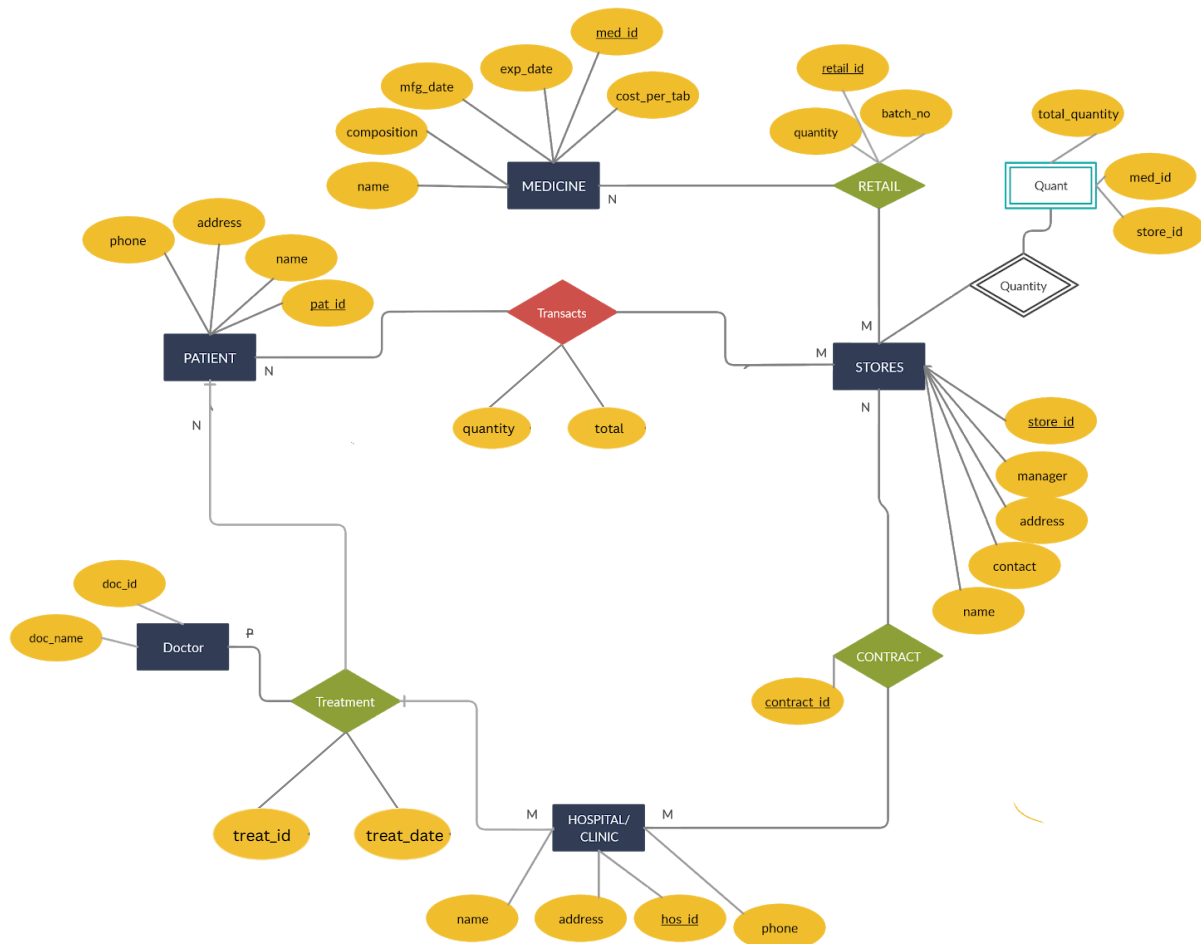


Fig 1: ER diagram of Pharmaceutical-Retail Database System

- All the strong(regular) entities, in Black, have a corresponding relation of their own - MEDICINE, STORES,, PATIENT, DOCTOR and HOSPITAL.
- The entire process of a PATIENT transacting with a STORE is modeled into a single relation called TRANSACTION. The ERD elaborates what is involved in a TRANSACTION.
- RETAIL - It is a relationship between the medicine(*med\_id*) and the store(*store\_id*). The medicines(*med\_id*) and the batch number(*batch\_no*) along with quantity supplied is noted down.

## Pharmaceutical - Retail Management System

- 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.
- Each relation is mapped using an appropriate foreign key keeping in mind the cardinality and participation constraints.

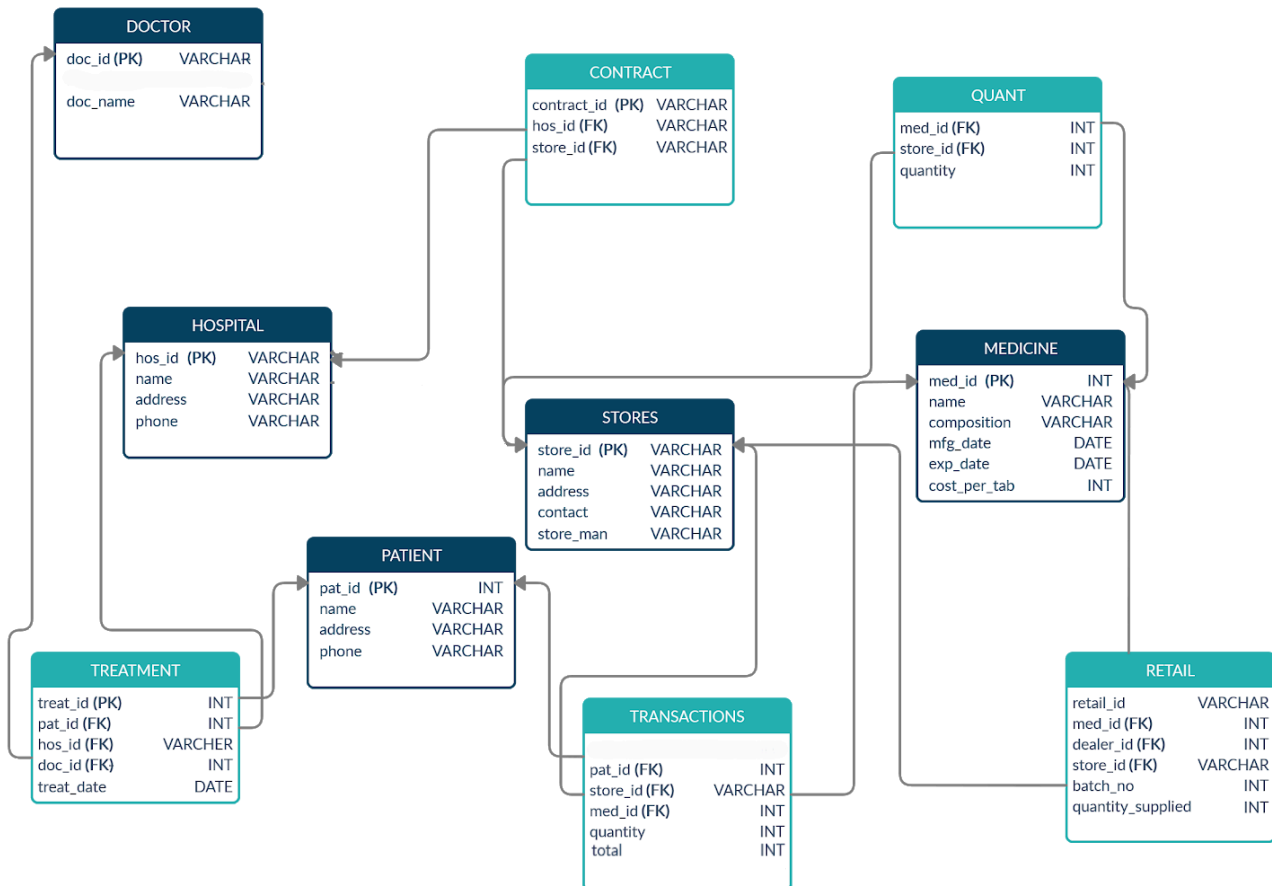


Fig 2: Relational Schema of Pharmaceutical-Retail Database System

The attributes along with its data types of all the relations can be seen in Fig 2. Some id's like *store\_id* and *hos\_id* are chosen to be of type VARCHAR while the other id's are INT. DATE type has been used in attributes whose values are dates of the calendar in "yyyy-mm-dd" format.

## 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.

## Second Normal Form (2NF):

HOSPITAL	
hos_id (PK)	INT
store_id (FK)	INT
name	VARCHAR
address	VARCHAR
phone	VARCHAR

Fig 3: HOSPITAL relation not in 2NF

CONTRACT	
contract_id (PK)	VARCHAR
hos_id (FK)	VARCHAR
store_id (FK)	VARCHAR

HOSPITAL	
hos_id (PK)	VARCHAR
name	VARCHAR
address	VARCHAR
phone	VARCHAR

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 :

- $hos\_id \rightarrow \{store\_id, name, address, phone\}$
- $hos\_id \rightarrow \{name, address, phone\}$
- $(name, address) \rightarrow \{hos\_id, phone\}$

But in no way does store\_id determine other attributes of HOSPITAL excluding hos\_id i.e  $store\_id \rightarrow \{name, address, phone\}$  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  $hos\_id \rightarrow \{name, address, phone\}$  and  $(name, address) \rightarrow \{hos\_id, phone\}$ . In CONTRACT table  $contract\_id \rightarrow \{hos\_id, store\_id\}$  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.

### Third Normal Form (3NF):

Earlier in the schema design , Fig 5, shows the relation TREATMENT. <Excluded 'type' attribute later on>

TREATMENT	
pat_id (FK)	INT
hos_id (FK)	INT
doc_name	VARCHAR

Fig 5: TREATMENT relation not in 3NF

TREATMENT	
treat_id (PK)	INT
pat_id (FK)	INT
hos_id (FK)	VARCHAR
doc_id (FK)	INT
treat_date	DATE

DOCTOR	
doc_id (PK)	VARCHAR
hos_id (FK)	VARCHAR

Fig 6: Solution to transitive dependency in Fig 5

The dependencies were  $pat\_id \rightarrow \{hos\_id\}$  and  $hos\_id \rightarrow \{doc\_name\}$ . 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.

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



## ➤ Data Definition Language (DDL) commands

---

DDL commands are used to define the database schema. It deals with descriptions of the database schema and is used to create and modify the structure of database objects in the database. Commands include CREATE, DROP, ALTER, COMMENT etc.

```
CREATE TABLE MEDICINE
```

```
(  
    med_id INT NOT NULL PRIMARY KEY ,  
    name VARCHAR(25) NOT NULL,  
    composition VARCHAR(70),  
    mfg_date DATE NOT NULL,  
    exp_date DATE NOT NULL,  
    cost_per_tab INT  
);
```

```
CREATE TABLE STORES
```

```
(  
    store_id VARCHAR(20) NOT NULL PRIMARY KEY,  
    name VARCHAR(50) NOT NULL,  
    address VARCHAR(70),  
    contact VARCHAR(25) NOT NULL,  
    store_man VARCHAR(25) NOT NULL  
);
```

```
CREATE TABLE HOSPITAL
```

```
(  
    hos_id VARCHAR(20) NOT NULL,  
    name VARCHAR(25) NOT NULL,  
    address VARCHAR(70),  
    phone VARCHAR(25) NOT NULL,  
    CONSTRAINT HOSPITAL_PK PRIMARY KEY(hos_id)  
);
```

```
CREATE TABLE CONTRACT
```

```
(  
    contract_id VARCHAR(20) NOT NULL,  
    hos_id VARCHAR(20) NOT NULL,  
    store_id VARCHAR(20) NOT NULL,  
    CONSTRAINT CONTRACT_PK PRIMARY KEY(hos_id, store_id),  
    CONSTRAINT CONTRACT_hos_id_FK FOREIGN KEY(hos_id) REFERENCES HOSPITAL(hos_id),  
    CONSTRAINT CONTRACT_store_id_FK FOREIGN KEY(store_id) REFERENCES STORES(store_id)  
);
```

## Pharmaceutical - Retail Management System

CREATE TABLE DOCTOR

```
(
  doc_id INT NOT NULL,
  doc_name VARCHAR(25) NOT NULL,
  CONSTRAINT DOCTOR_PK PRIMARY KEY(doc_id),
);
```

CREATE TABLE RETAIL

```
(
  retail_id VARCHAR(20) NOT NULL,
  med_id INT NOT NULL,
  store_id VARCHAR(20) NOT NULL,
  batchno INT NOT NULL,
  quantity_supplied INT NOT NULL,
  CONSTRAINT RETAIL_PK PRIMARY KEY(retail_id, med_id, store_id),
  CONSTRAINT RETAIL_med_id_FK FOREIGN KEY(med_id) REFERENCES MEDICINE(med_id),
  CONSTRAINT RETAIL_store_id_FK FOREIGN KEY(store_id) REFERENCES STORES(store_id),
);
```

CREATE TABLE PATIENT

```
(
  pat_id INT NOT NULL PRIMARY KEY,
  name VARCHAR(25) NOT NULL,
  address VARCHAR(70),
  phone VARCHAR(15) NOT NULL
);
```

CREATE TABLE TREATMENT

```
(
  treat_id INT NOT NULL UNIQUE,
  pat_id INT NOT NULL,
  hos_id VARCHAR(20) NOT NULL,
  doc_id INT NOT NULL,
  treat_date DATE NOT NULL,
  CONSTRAINT TREATMENT_PK PRIMARY KEY(treat_id, pat_id, hos_id, doc_id),
  CONSTRAINT TREATMENT_pat_id_FK FOREIGN KEY(pat_id) REFERENCES PATIENT(pat_id),
  CONSTRAINT TREATMENT_hos_id_FK FOREIGN KEY(hos_id) REFERENCES HOSPITAL(hos_id),
  CONSTRAINT TREATMENT_doc_id_FK FOREIGN KEY(doc_id) REFERENCES DOCTOR(doc_id)
);
```

CREATE TABLE TRANSACTIONS

```
(
  pat_id INT NOT NULL,
  store_id VARCHAR(20) NOT NULL,
  med_id INT NOT NULL,
  quantity INT NOT NULL,
  total INT,
  CONSTRAINT TRANSACTIONS_PK PRIMARY KEY(pat_id, med_id, store_id),
  CONSTRAINT TRANSACTIONS_pat_id_FK FOREIGN KEY(pat_id) REFERENCES PATIENT(pat_id),
  CONSTRAINT TRANSACTIONS_med_id_FK FOREIGN KEY(med_id) REFERENCES MEDICINE(med_id),
  CONSTRAINT TRANSACTIONS_store_id_FK FOREIGN KEY(store_id) REFERENCES STORES(store_id)
);
```

## Pharmaceutical - Retail Management System

```
CREATE TABLE QUANT
(
  med_id INT ,
  store_id VARCHAR(20) ,
  quantity INT,
  CONSTRAINT QUANT_PK PRIMARY KEY(med_id,store_id),
  CONSTRAINT QUANT_med_id_FK FOREIGN KEY(med_id) REFERENCES MEDICINE(med_id),
  CONSTRAINT QUANT_store_id_FK FOREIGN KEY(store_id) REFERENCES STORES(store_id)
);
```

---

## Pharmaceutical - Retail Management System

```
INSERT ALL
INTO MEDICINE (med_id, name, composition, mfg_date, exp_date, cost_per_tab) VALUES (1, 'Crocini', 'paracetamol, caffeine', DATE '2019-12-24', DATE '2022-04-24', 9)
INTO MEDICINE (med_id, name, composition, mfg_date, exp_date, cost_per_tab) VALUES (2, 'Saridon', 'propyphenazone, paracetamol, caffeine', DATE '2019-04-15', DATE '2024-03-10', 5)
INTO MEDICINE (med_id, name, composition, mfg_date, exp_date, cost_per_tab) VALUES (3, 'Vicks', 'camphor, nutmeg, thymol, cedarwood', DATE '2018-11-02', DATE '2022-06-27', 6)
INTO MEDICINE (med_id, name, composition, mfg_date, exp_date, cost_per_tab) VALUES (4, 'Gelusil', 'aluminum hydroxide, magnesium hydroxide', DATE '2018-07-09', DATE '2023-06-20', 12)
INTO MEDICINE (med_id, name, composition, mfg_date, exp_date, cost_per_tab) VALUES (5, 'Mosha-L', 'levocetirizine, montelukast', DATE '2019-01-06', DATE '2022-07-30', 7)
INTO MEDICINE (med_id, name, composition, mfg_date, exp_date, cost_per_tab) VALUES (6, 'Glycomet', 'metformin', DATE '2020-06-06', DATE '2024-01-24', 15)
INTO MEDICINE (med_id, name, composition, mfg_date, exp_date, cost_per_tab) VALUES (7, 'Dolo650', 'paracetamol', DATE '2018-12-15', DATE '2022-02-17', 8)
INTO MEDICINE (med_id, name, composition, mfg_date, exp_date, cost_per_tab) VALUES (8, 'Volini', 'diclofenac diethylamine, methanol, linseed oil', DATE '2018-12-31', DATE '2023-08-13', 2)
INTO MEDICINE (med_id, name, composition, mfg_date, exp_date, cost_per_tab) VALUES (9, 'Benadryl', 'diphenhydramine hydrochloride, ammonium chloride', DATE '2019-12-23', DATE '2022-06-12', 50)
INTO MEDICINE (med_id, name, composition, mfg_date, exp_date, cost_per_tab) VALUES (10, 'Otrivin', 'xylometazoline hydrochloride', DATE '2020-11-16', DATE '2023-11-11', 86)
INTO MEDICINE (med_id, name, composition, mfg_date, exp_date, cost_per_tab) VALUES (11, 'Ciplox', 'quinolone antibiotics', DATE '2019-06-14', DATE '2022-09-20', 99)
INTO MEDICINE (med_id, name, composition, mfg_date, exp_date, cost_per_tab) VALUES (12, 'Eldoper', 'Loperamide', DATE '2020-02-15', DATE '2023-08-24', 5)
SELECT * FROM DUAL;
```

```
SELECT * FROM MEDICINE
```

```
INSERT ALL
INTO STORES (store_id, name, address, contact, store_man) VALUES ('S001', 'Super Mart', '123 Main St, Cityville', '123-456-7890', 'John Doe')
INTO STORES (store_id, name, address, contact, store_man) VALUES ('S002', 'Mega Mall', '456 Center Ave, Townsville', '987-654-3210', 'Jane Smith')
INTO STORES (store_id, name, address, contact, store_man) VALUES ('S003', 'Quick Stop', '789 Elm St, Villagetown', '555-123-4567', 'Mike Johnson')
INTO STORES (store_id, name, address, contact, store_man) VALUES ('S004', 'Corner Store', '321 Oak St, Hamletville', '111-222-3333', 'Sarah Lee')
INTO STORES (store_id, name, address, contact, store_man) VALUES ('S005', 'Shop-N-Go', '654 Pine St, Suburbia', '444-555-6666', 'David Brown')
INTO STORES (store_id, name, address, contact, store_man) VALUES ('S006', 'City Market', '789 Maple Ave, Downtown', '777-888-9999', 'Emily White')
INTO STORES (store_id, name, address, contact, store_man) VALUES ('S007', 'Fresh Fare', '987 Cedar St, Uptown', '222-333-4444', 'Michael Green')
INTO STORES (store_id, name, address, contact, store_man) VALUES ('S008', 'Farmers Corner', '654 Rose St, Countryside', '888-999-0000', 'Jessica Taylor')
INTO STORES (store_id, name, address, contact, store_man) VALUES ('S009', 'Town Grocers', '123 Magnolia St, Town Center', '333-444-5555', 'William Clark')
INTO STORES (store_id, name, address, contact, store_man) VALUES ('S010', 'Neighborhood Market', '456 Birch St, Suburbia', '666-777-8888', 'Laura Martinez')
SELECT * FROM DUAL;
```

```
select * from STORES
```

```
INSERT ALL
INTO HOSPITAL (hos_id, name, address, phone) VALUES ('H001', 'City General', '123 Main St, Cityville', '123-456-7890')
INTO HOSPITAL (hos_id, name, address, phone) VALUES ('H002', 'Metro Health', '456 Center Ave, Townsville', '987-654-3210')
INTO HOSPITAL (hos_id, name, address, phone) VALUES ('H003', 'Community Hospital', '789 Elm St, Villagetown', '555-123-4567')
INTO HOSPITAL (hos_id, name, address, phone) VALUES ('H004', 'Unity Medical Center', '321 Oak St, Hamletville', '111-222-3333')
INTO HOSPITAL (hos_id, name, address, phone) VALUES ('H005', 'County Hospital', '654 Pine St, Suburbia', '444-555-6666')
INTO HOSPITAL (hos_id, name, address, phone) VALUES ('H006', 'Saint Joseph Hospital', '789 Maple Ave, Downtown', '777-888-9999')
INTO HOSPITAL (hos_id, name, address, phone) VALUES ('H007', 'Westside Medical Center', '987 Cedar St, Uptown', '222-333-4444')
INTO HOSPITAL (hos_id, name, address, phone) VALUES ('H008', 'Valley Hospital', '654 Rose St, Countryside', '888-999-0000')
INTO HOSPITAL (hos_id, name, address, phone) VALUES ('H009', 'Northside Hospital', '123 Magnolia St, Town Center', '333-444-5555')
INTO HOSPITAL (hos_id, name, address, phone) VALUES ('H010', 'University Hospital', '456 Birch St, Suburbia', '666-777-8888')
SELECT * FROM DUAL
```

```
SELECT * FROM HOSPITAL
```

## Pharmaceutical - Retail Management System

INSERT ALL

```
INTO CONTRACT (contract_id, hos_id, store_id) VALUES ('C001', 'H001', 'S001')
INTO CONTRACT (contract_id, hos_id, store_id) VALUES ('C002', 'H002', 'S002')
INTO CONTRACT (contract_id, hos_id, store_id) VALUES ('C003', 'H003', 'S003')
INTO CONTRACT (contract_id, hos_id, store_id) VALUES ('C004', 'H004', 'S004')
INTO CONTRACT (contract_id, hos_id, store_id) VALUES ('C005', 'H005', 'S005')
INTO CONTRACT (contract_id, hos_id, store_id) VALUES ('C006', 'H006', 'S006')
INTO CONTRACT (contract_id, hos_id, store_id) VALUES ('C007', 'H007', 'S007')
INTO CONTRACT (contract_id, hos_id, store_id) VALUES ('C008', 'H008', 'S008')
INTO CONTRACT (contract_id, hos_id, store_id) VALUES ('C009', 'H009', 'S009')
INTO CONTRACT (contract_id, hos_id, store_id) VALUES ('C010', 'H010', 'S010')
SELECT * FROM DUAL
```

SELECT \* FROM CONTRACT

INSERT ALL

```
INTO DOCTOR (doc_id, doc_name) VALUES (1, 'Dr. Smith')
INTO DOCTOR (doc_id, doc_name) VALUES (2, 'Dr. Johnson')
INTO DOCTOR (doc_id, doc_name) VALUES (3, 'Dr. Williams')
INTO DOCTOR (doc_id, doc_name) VALUES (4, 'Dr. Brown')
INTO DOCTOR (doc_id, doc_name) VALUES (5, 'Dr. Jones')
INTO DOCTOR (doc_id, doc_name) VALUES (6, 'Dr. Garcia')
INTO DOCTOR (doc_id, doc_name) VALUES (7, 'Dr. Martinez')
INTO DOCTOR (doc_id, doc_name) VALUES (8, 'Dr. Anderson')
INTO DOCTOR (doc_id, doc_name) VALUES (9, 'Dr. Taylor')
INTO DOCTOR (doc_id, doc_name) VALUES (10, 'Dr. Thomas')
SELECT * FROM DUAL
```

SELECT \* FROM DOCTOR

INSERT ALL

```
INTO PATIENT (pat_id, name, address, phone) VALUES (1, 'John Smith', '123 Main St, Cityville', '123-456-7890')
INTO PATIENT (pat_id, name, address, phone) VALUES (2, 'Jane Doe', '456 Center Ave, Townsville', '987-654-3210')
INTO PATIENT (pat_id, name, address, phone) VALUES (3, 'Michael Johnson', '789 Elm St, Villagetown', '555-123-4567')
INTO PATIENT (pat_id, name, address, phone) VALUES (4, 'Sarah Brown', '321 Oak St, Hamletville', '111-222-3333')
INTO PATIENT (pat_id, name, address, phone) VALUES (5, 'Emily Davis', '654 Pine St, Suburbia', '444-555-6666')
INTO PATIENT (pat_id, name, address, phone) VALUES (6, 'David Wilson', '789 Maple Ave, Downtown', '777-888-9999')
INTO PATIENT (pat_id, name, address, phone) VALUES (7, 'Jessica Lee', '987 Cedar St, Uptown', '222-333-4444')
INTO PATIENT (pat_id, name, address, phone) VALUES (8, 'Matthew Clark', '654 Rose St, Countryside', '888-999-0000')
INTO PATIENT (pat_id, name, address, phone) VALUES (9, 'Laura Taylor', '123 Magnolia St, Town Center', '333-444-5555')
INTO PATIENT (pat_id, name, address, phone) VALUES (10, 'Daniel Martinez', '456 Birch St, Suburbia', '666-777-8888')
SELECT * FROM DUAL
```

SELECT \* FROM PATIENT

INSERT ALL

```
INTO TREATMENT (treat_id, pat_id, hos_id, doc_id, treat_date) VALUES (1, 1, 'H001', 1, TO_DATE('2024-03-30', 'YYYY-MM-DD'))
INTO TREATMENT (treat_id, pat_id, hos_id, doc_id, treat_date) VALUES (2, 2, 'H002', 2, TO_DATE('2024-03-31', 'YYYY-MM-DD'))
INTO TREATMENT (treat_id, pat_id, hos_id, doc_id, treat_date) VALUES (3, 3, 'H003', 3, TO_DATE('2024-04-01', 'YYYY-MM-DD'))
INTO TREATMENT (treat_id, pat_id, hos_id, doc_id, treat_date) VALUES (4, 4, 'H004', 4, TO_DATE('2024-04-02', 'YYYY-MM-DD'))
INTO TREATMENT (treat_id, pat_id, hos_id, doc_id, treat_date) VALUES (5, 5, 'H005', 5, TO_DATE('2024-04-03', 'YYYY-MM-DD'))
INTO TREATMENT (treat_id, pat_id, hos_id, doc_id, treat_date) VALUES (6, 6, 'H006', 6, TO_DATE('2024-04-04', 'YYYY-MM-DD'))
INTO TREATMENT (treat_id, pat_id, hos_id, doc_id, treat_date) VALUES (7, 7, 'H007', 7, TO_DATE('2024-04-05', 'YYYY-MM-DD'))
INTO TREATMENT (treat_id, pat_id, hos_id, doc_id, treat_date) VALUES (8, 8, 'H008', 8, TO_DATE('2024-04-06', 'YYYY-MM-DD'))
INTO TREATMENT (treat_id, pat_id, hos_id, doc_id, treat_date) VALUES (9, 9, 'H009', 9, TO_DATE('2024-04-07', 'YYYY-MM-DD'))
INTO TREATMENT (treat_id, pat_id, hos_id, doc_id, treat_date) VALUES (10, 10, 'H010', 10, TO_DATE('2024-04-08', 'YYYY-MM-DD'))
SELECT * FROM DUAL
```

SELECT \* FROM TREATMENT

## Pharmaceutical - Retail Management System

```
INSERT ALL
INTO RETAIL (retail_id, med_id, store_id, batchno, quantity_supplied) VALUES ('R001', 1, 'S001', 1234, 100)
INTO RETAIL (retail_id, med_id, store_id, batchno, quantity_supplied) VALUES ('R002', 2, 'S002', 5678, 150)
INTO RETAIL (retail_id, med_id, store_id, batchno, quantity_supplied) VALUES ('R003', 3, 'S003', 91011, 200)
INTO RETAIL (retail_id, med_id, store_id, batchno, quantity_supplied) VALUES ('R004', 4, 'S004', 121314, 120)
INTO RETAIL (retail_id, med_id, store_id, batchno, quantity_supplied) VALUES ('R005', 5, 'S005', 151617, 180)
INTO RETAIL (retail_id, med_id, store_id, batchno, quantity_supplied) VALUES ('R006', 6, 'S006', 181920, 90)
INTO RETAIL (retail_id, med_id, store_id, batchno, quantity_supplied) VALUES ('R007', 7, 'S007', 212223, 220)
INTO RETAIL (retail_id, med_id, store_id, batchno, quantity_supplied) VALUES ('R008', 8, 'S008', 242526, 130)
INTO RETAIL (retail_id, med_id, store_id, batchno, quantity_supplied) VALUES ('R009', 9, 'S009', 272829, 170)
INTO RETAIL (retail_id, med_id, store_id, batchno, quantity_supplied) VALUES ('R010', 10, 'S010', 303132, 200)
SELECT * FROM DUAL
```

```
SELECT * FROM RETAIL
```

```
INSERT ALL
INTO TRANSACTIONS (pat_id, store_id, med_id, quantity, total) VALUES (1, 'S001', 1, 5, 50)
INTO TRANSACTIONS (pat_id, store_id, med_id, quantity, total) VALUES (2, 'S002', 2, 3, 30)
INTO TRANSACTIONS (pat_id, store_id, med_id, quantity, total) VALUES (3, 'S003', 3, 7, 70)
INTO TRANSACTIONS (pat_id, store_id, med_id, quantity, total) VALUES (4, 'S004', 4, 4, 40)
INTO TRANSACTIONS (pat_id, store_id, med_id, quantity, total) VALUES (5, 'S005', 5, 6, 60)
INTO TRANSACTIONS (pat_id, store_id, med_id, quantity, total) VALUES (6, 'S006', 6, 2, 20)
INTO TRANSACTIONS (pat_id, store_id, med_id, quantity, total) VALUES (7, 'S007', 7, 8, 80)
INTO TRANSACTIONS (pat_id, store_id, med_id, quantity, total) VALUES (8, 'S008', 8, 5, 50)
INTO TRANSACTIONS (pat_id, store_id, med_id, quantity, total) VALUES (9, 'S009', 9, 6, 60)
INTO TRANSACTIONS (pat_id, store_id, med_id, quantity, total) VALUES (10, 'S010', 10, 3, 30)
SELECT * FROM DUAL
```

```
SELECT * FROM TRANSACTIONS
```

```
INSERT ALL
INTO QUANT (med_id, store_id, quantity) VALUES (1, 'S001', 100)
INTO QUANT (med_id, store_id, quantity) VALUES (2, 'S002', 150)
INTO QUANT (med_id, store_id, quantity) VALUES (3, 'S003', 200)
INTO QUANT (med_id, store_id, quantity) VALUES (4, 'S004', 120)
INTO QUANT (med_id, store_id, quantity) VALUES (5, 'S005', 180)
INTO QUANT (med_id, store_id, quantity) VALUES (6, 'S006', 90)
INTO QUANT (med_id, store_id, quantity) VALUES (7, 'S007', 220)
INTO QUANT (med_id, store_id, quantity) VALUES (8, 'S008', 130)
INTO QUANT (med_id, store_id, quantity) VALUES (9, 'S009', 170)
INTO QUANT (med_id, store_id, quantity) VALUES (10, 'S010', 200)
SELECT * FROM DUAL
```

```
SELECT * FROM QUANT
```

## ➤ Sample Queries

### 1. What are the details of treatments conducted by Dr. Smith?

```
SELECT t.treat_id, p.name AS patient_name, h.name AS hospital_name, t.treat_date
FROM TREATMENT t
JOIN PATIENT p ON t.pat_id = p.pat_id
JOIN HOSPITAL h ON t.hos_id = h.hos_id
JOIN DOCTOR d ON t.doc_id = d.doc_id
WHERE d.doc_name = 'Dr. Smith';
```

TREAT_ID	PATIENT_NAME	HOSPITAL_NAME	TREAT_DATE
1	John Smith	City General	30-MAR-24

### 2. How many units of each medicine are available in each store?

```
SELECT q.med_id, m.name AS medicine_name, q.store_id, s.name AS store_name, q.quantity
FROM QUANT q
JOIN MEDICINE m ON q.med_id = m.med_id
JOIN STORES s ON q.store_id = s.store_id;
```

MED_ID	MEDICINE_NAME	STORE_ID	STORE_NAME	QUANTITY
1	Crocin	S001	Super Mart	100
2	Saridon	S002	Mega Mall	150
3	Vicks	S003	Quick Stop	200
4	Gelusil	S004	Corner Store	120
5	Mosha-L	S005	Shop-N-Go	180
6	Glycomet	S006	City Market	90
7	Dolo650	S007	Fresh Fare	220
8	Volini	S008	Farmers Corner	130
9	Benadryl	S009	Town Grocers	170
10	Otrivin	S010	Neighborhood Market	200

## ➤ Sample Queries

### 3. What are the transactions made by patients living in Suburbia?

```
SELECT t.pat_id, p.name AS patient_name, t.med_id, m.name AS medicine_name, t.store_id, s.name AS store_name, t.quantity, t.total
FROM TRANSACTIONS t
JOIN PATIENT p ON t.pat_id = p.pat_id
JOIN STORES s ON t.store_id = s.store_id
JOIN MEDICINE m ON t.med_id = m.med_id
WHERE p.address LIKE '%Suburbia%';
```

PAT_ID	PATIENT_NAME	MED_ID	MEDICINE_NAME	STORE_ID	STORE_NAME	QUANTITY	TOTAL
5	Emily Davis	5	Mosha-L	S005	Shop-N-Go	6	60
10	Daniel Martinez	10	Otrivin	S010	Neighborhood Market	3	30

### 4. Which medicines are going to expire soon (within the next three months)?

```
SELECT med_id, name AS medicine_name, exp_date
FROM MEDICINE
WHERE exp_date BETWEEN SYSDATE AND ADD_MONTHS(SYSDATE, 3);
```

MED_ID	MEDICINE_NAME	EXP_DATE
2	Saridon	30-JUN-24



## ➤ Conclusion

---

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 from 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.