Database and Management Systems

(UEC716)

Lab Project

on

Health Hub

Hospital Database Management System

THAPAR INSTITUTE
OF ENGINEERING & TECHNOLOGY
(Deemed to be University)

Submitted by:

Pranav Bakshi	102215220
Krish Juneja	102215237
Prerna Nagpal	102215243
Angad Singh	102215345

DEPARTMENT OF ELECTRONICS AND COMMUNICATION ENGINEERING

THAPAR INSTITUTE OF ENGINEERING & TECHNOLOGY PATIALA

August-December 2024

Table of Contents

REQUIREMENTS

The doctor-patient management system is a software application that is designed to manage the interactions between doctors and patients. The system allows doctors to manage their appointments, prescriptions, and medical records for their patients. Patients can use the system to schedule appointments, view their medical records, and receive prescriptions.

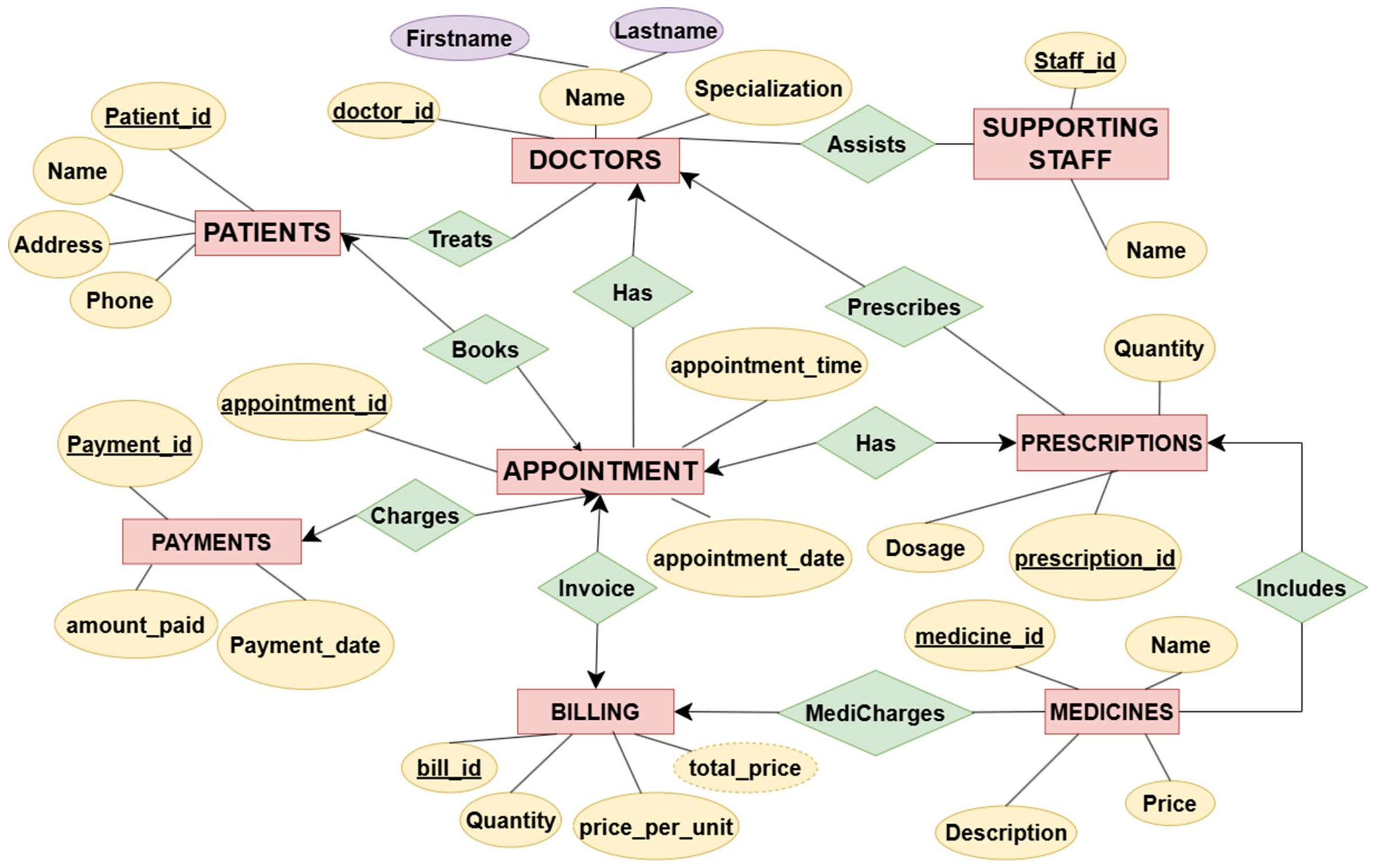
The system is implemented using a database to store the doctor, patient, appointment, prescription, billing, and payment information. The system also includes a user interface that allows doctors and patients to interact with the database.

The doctor-patient management system can be used in a variety of healthcare settings, such as hospitals, clinics, and private practices. The system can improve the efficiency and accuracy of managing patient information, reduce errors, and enhance communication between doctors and patients.

To commercialize the system, it can be marketed to healthcare organizations and private practices. The system can be sold as a subscription-based service or a one-time purchase. The system can also be customized to meet the specific needs of healthcare organizations and practices, which can provide an additional source of revenue.

In summary, the doctor-patient management system is a valuable tool for managing patient information, improving communication, and enhancing the overall quality of healthcare. It can be implemented in various healthcare settings and can be commercialized through subscriptions or customized solutions for organizations and practices.

MODELLING OF REQUIREMENTS AS ER-DIAGRAM



ASSUMPTIONS:

A Doctor can have many Appointments, but each Appointment is with only one Doctor.
A Patient can have only one Appointment and each Appointment has only one Patient.
A Prescription is associated with one Appointment and can have many Medicines.
Many Medicines can be associated with one Prescription and one Billing.
Many supporting staff assist many Doctors.
A Billing is associated with one Appointment and can have many Medicines.
A Payment is associated with one Appointment.

ER- DIAGRAM INTO RELATIONAL SCHEMA

Doctors

doctor_id	First Name	Last Name	Specialization

DOCTORS(doctor_id , Firstname , Lastname , Specialization)

Patients

Patient_id	Name	Address	Phone

PATIENTS(Patients_id , name , Address, Phone)

Billing

BILLI
NG(

Bill_id	Quantity	Price per unit	Total price	Prescription_id

Bill_id , Quantity , price_per_unit , Total_price,prescription_id)

Medicines

Medicine_id	Name	Description	Price

MEDICINES(Medicines_id , Name , Description , price)

Supporting Staff

Supporting_id	Name

SUPPORTING STAFF(Staff_id , Name)

Appointments

Appointment_id	Appointment_Date	Appointment_time	Patient_id	Doctor_id

APPOINTMENTS(Appointment_id,Appointment_Date, Appointment_time,Patient_id,Doctor_id)

Prescriptions

Prescription_id	Dosage	Quantity	Appointment_id	Medicine_id

PRESCRIPTION(Prescription_id,Dosage,Quantity,Appointment_id,Medicine_id)

AFTER NORMALIZATION

Doctors

doctor_id	First name	Last name	Specialization

Primary Key: doctor _id
Foreign Keys: staff_id

Patients

patient_id	name	address

· Primary Key: patient_id

patient_id	Phone

Primary Key: Phone
Foreign Key: patient_id

Appointments

appointment_id	doctor_id	patient_id	appointment_date	appointment_time

Primary Key: appointment_id
Foreign Keys: doctor_id ,patient_id

Medicines

medicine_id	name	description	price	Bill_id

Primary Key: medicine _id
Foreign Keys: bill_id

Prescriptions

prescription_id	appointment_id	medicine_id	dosage	quantity

Primary Key: prescription_id
Foreign Keys:appointment_id, medicine_id

Billing

bill_id	appointment_id	medicine_id	quantity	price_per_unit	total_price

Primary Key: bill_id
Foreign Keys: appointment_id, medicine_id

Payments

payment_id	appointment_id	amount_paid	payment_date

· Primary Key: payment_id
· Foreign Keys:appointment_id

Supporting_Staff

staff _id	name

· Primary Key: staff_id

SQL STATEMENTS FOR TABLE CREATION

Creating DOCTORS Table

CREATE TABLE Doctors (doctor_id NUMBER(10) PRIMARY KEY,
name VARCHAR2(50) NOT NULL,
specialization VARCHAR2(50) NOT NULL);

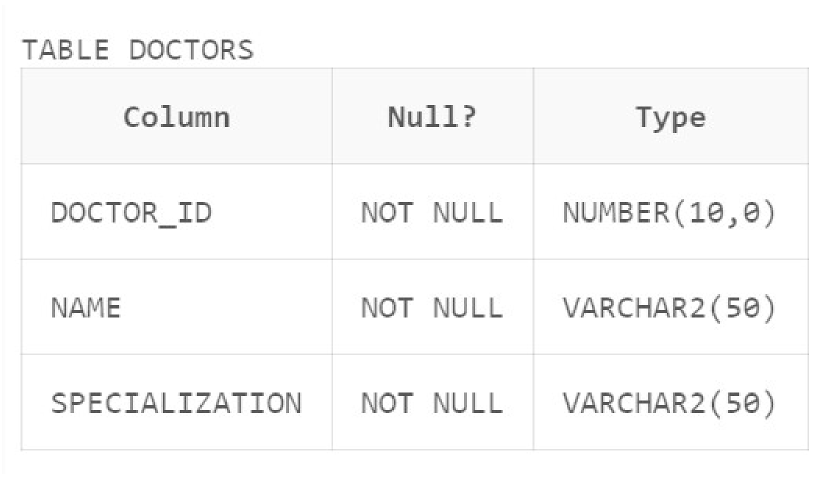


TABLE DOCTORS

Column	Null?	Type
DOCTOR_ID	NOT NULL	NUMBER(10,0)
NAME	NOT NULL	VARCHAR2(50)
SPECIALIZATION	NOT NULL	VARCHAR2(50)

Creating Patients Table

CREATE TABLE Patients (patient_id

NUMBER(10) PRIMARY KEY,

nameVARCHAR2(50) NOT NULL,

addressVARCHAR2(100) NOT NULL,

phoneVARCHAR2(20) NOT NULL);

Creating Appointments Table

```
CREATE TABLE Appointments ( appointment_id NUMBER(10) PRIMARY KEY,

doctor_id NUMBER(10) REFERENCES Doctors(doctor_id),

patient_id NUMBER(10) REFERENCES Patients(patient_id),

appointment_date DATE NOT NULL,

appointment_timeVARCHAR2(20) NOT NULL

    );
```

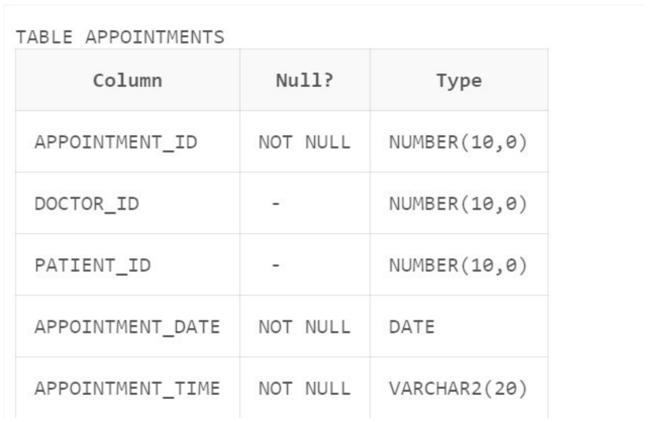


TABLE APPOINTMENTS

Column	Null?	Type
APPOINTMENT_ID	NOT NULL	NUMBER(10,0)
DOCTOR_ID	-	NUMBER(10,0)
PATIENT_ID	-	NUMBER(10,0)
APPOINTMENT_DATE	NOT NULL	DATE
APPOINTMENT_TIME	NOT NULL	VARCHAR2(20)

Creating Medicines Table

```
CREATE TABLE Medicines ( medicine_id

NUMBER(10) PRIMARY KEY,


Name VARCHAR2(50) NOT NULL,

description VARCHAR2(100) NOT NULL,

price NUMBER(10, 2) NOT NULL

    );
```

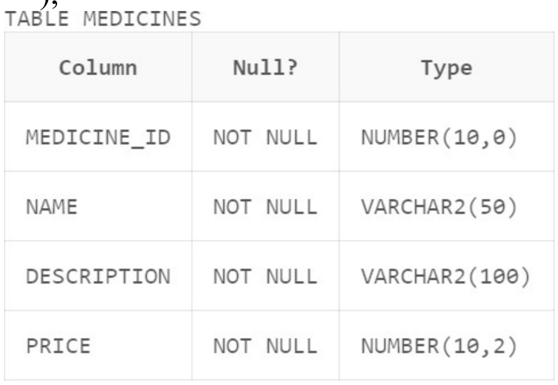


TABLE MEDICINES

Column	Null?	Type
MEDICINE_ID	NOT NULL	NUMBER(10,0)
NAME	NOT NULL	VARCHAR2(50)
DESCRIPTION	NOT NULL	VARCHAR2(100)
PRICE	NOT NULL	NUMBER(10,2)

Creating Prescriptions Table

CREATE TABLE Prescriptions (

prescription_id NUMBER(10) PRIMARYKEY,

appointment_id NUMBER(10) REFERENCES Appointments(appointment_id),

medicine_id NUMBER(10) REFERENCESMedicines(medicine_id),

dosage VARCHAR2(50) NOT NULL,

quantity NUMBER(10) NOT NULL);

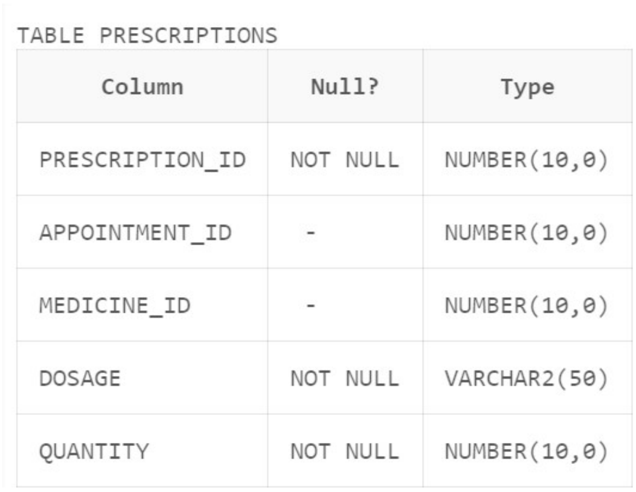


TABLE PRESCRIPTIONS

Column	Null?	Type
PRESCRIPTION_ID	NOT NULL	NUMBER(10,0)
APPOINTMENT_ID	-	NUMBER(10,0)
MEDICINE_ID	-	NUMBER(10,0)
DOSAGE	NOT NULL	VARCHAR2(50)
QUANTITY	NOT NULL	NUMBER(10,0)

Creating Billing Table

CREATE TABLE Billing (bill_id NUMBER(10) PRIMARY KEY,

appointment_id NUMBER(10) REFERENCES Appointments(appointment_id),

medicine_id NUMBER(10) REFERENCES Medicines(medicine_id), quantity

NUMBER(10) NOT NULL, price_per_unit NUMBER(10, 2) NOT NULL,

total_price NUMBER(10, 2) NOT NULL

);

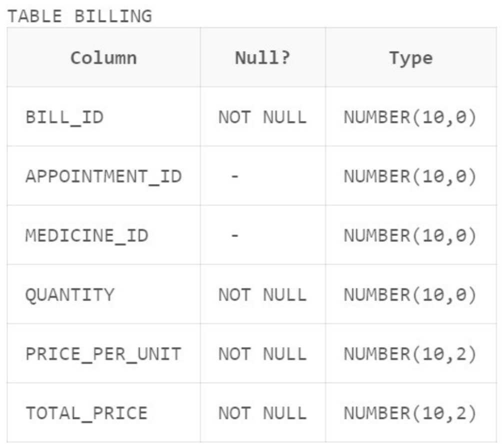


TABLE BILLING

Column	Null?	Type
BILL_ID	NOT NULL	NUMBER(10,0)
APPOINTMENT_ID	-	NUMBER(10,0)
MEDICINE_ID	-	NUMBER(10,0)
QUANTITY	NOT NULL	NUMBER(10,0)
PRICE_PER_UNIT	NOT NULL	NUMBER(10,2)
TOTAL_PRICE	NOT NULL	NUMBER(10,2)

Creating Payments Table

```
CREATE TABLE Payments ( payment_id NUMBER(10) PRIMARY KEY,

appointment_id NUMBER(10) REFERENCES Appointments(appointment_id),

amount_paid NUMBER(10, 2) NOT NULL, payment_date DATE NOT NULL

);
```

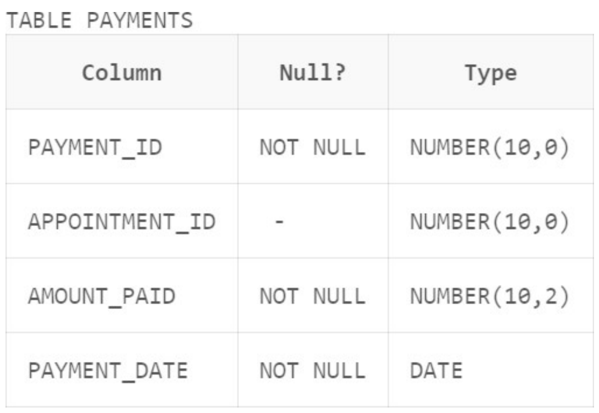
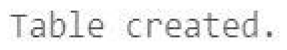


TABLE PAYMENTS

Column	Null?	Type
PAYMENT_ID	NOT NULL	NUMBER(10,0)
APPOINTMENT_ID	-	NUMBER(10,0)
AMOUNT_PAID	NOT NULL	NUMBER(10,2)
PAYMENT_DATE	NOT NULL	DATE

Creating Supporting Staff Table

```
Create table supporting_staff(staff_id number primary key, name varchar(30));
```

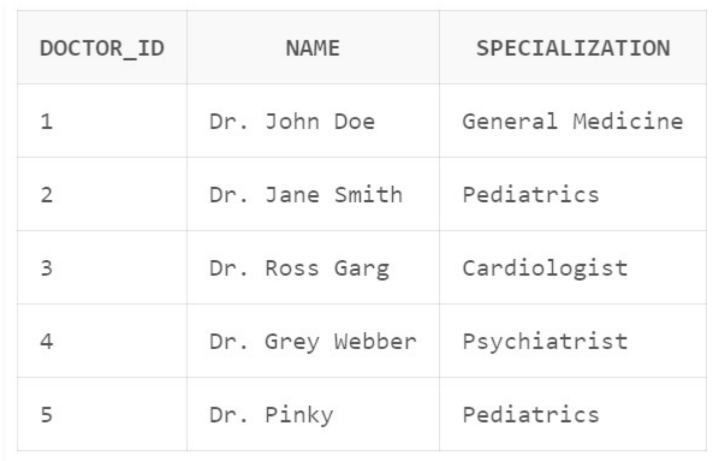


```
Table created.
```

STATEMENTS FOR INSERT COMMANDS

Inserting data into Doctors Table

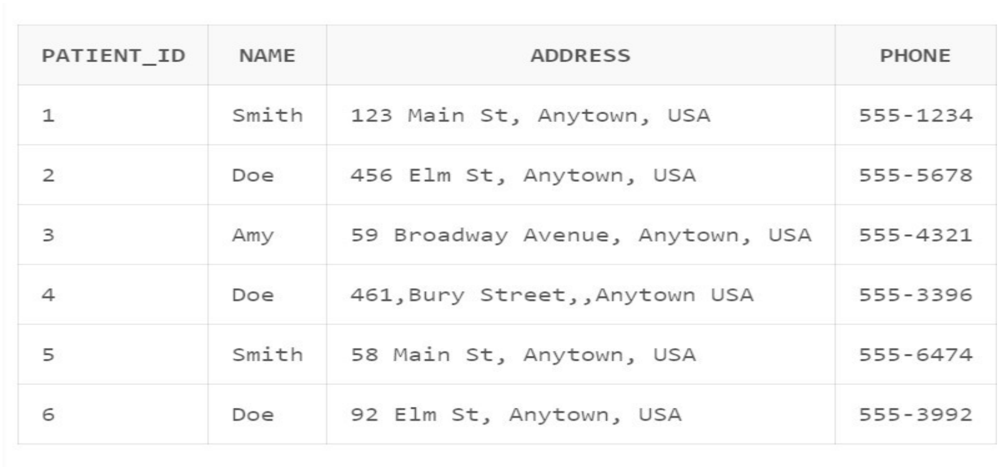
```
INSERT INTO Doctors (doctor_id, name, specialization) VALUES (1, 'Dr. John Doe', 'General Medicine');
INSERT INTO Doctors (doctor_id, name, specialization) VALUES (2, 'Dr. Jane Smith', 'Pediatrics');

INSERT INTO Doctors (doctor_id, name, specialization) VALUES (3, 'Dr. Ross Garg', 'Cardiologist');

INSERT INTO Doctors (doctor_id, name, specialization) VALUES (4, 'Dr. Grey Webber', 'Psychiatrist');

INSERT INTO Doctors (doctor_id, name, specialization) VALUES (5, 'Dr. Pinky', 'Pediatrics');
```



DOCTOR_ID	NAME	SPECIALIZATION
1	Dr. John Doe	General Medicine
2	Dr. Jane Smith	Pediatrics
3	Dr. Ross Garg	Cardiologist
4	Dr. Grey Webber	Psychiatrist
5	Dr. Pinky	Pediatrics

Inserting data into Patients Table

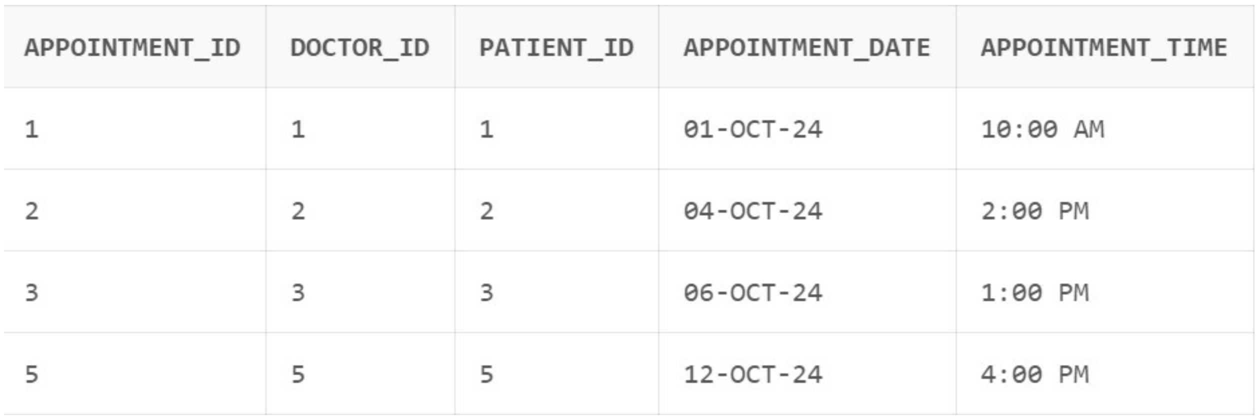
```
INSERT INTO Patients (patient_id, name, address, phone) VALUES (1, 'Smith', '123 Main St, Anytown,
USA', '555-1234');
INSERT INTO Patients (patient_id, name, address, phone) VALUES (2, 'Doe', '456 Elm St, Anytown,
USA', '555-5678');
INSERT INTO Patients (patient_id, name, address, phone) VALUES (3, 'Amy', '59 Broadway Avenue, Anytown,
USA', '555-4321');
INSERT INTO Patients (patient_id, name, address, phone) VALUES (2, 'Doe', '461,Bury Street,,Anytown
USA', '555-3396');
INSERT INTO Patients (patient_id, name, address, phone) VALUES (1, 'Smith', '58 Main St, Anytown,
USA', '555-6474');
INSERT INTO Patients (patient_id, name, address, phone) VALUES (2, 'Doe', '92 Elm St, Anytown, USA', '555-
3992');
```



PATIENT_ID	NAME	ADDRESS	PHONE
1	Smith	123 Main St, Anytown, USA	555-1234
2	Doe	456 Elm St, Anytown, USA	555-5678
3	Amy	59 Broadway Avenue, Anytown, USA	555-4321
4	Doe	461,Bury Street,,Anytown USA	555-3396
5	Smith	58 Main St, Anytown, USA	555-6474
6	Doe	92 Elm St, Anytown, USA	555-3992

Inserting data into Appointments Table

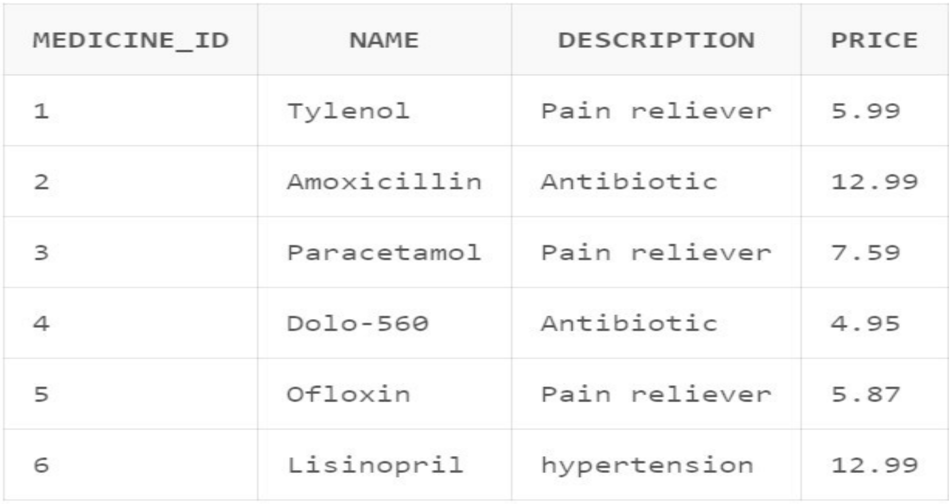
```
INSERT INTO Appointments (appointment_id, doctor_id, patient_id, appointment_date, appointment_time)
VALUES (1, 1, 1, TO_DATE('2024-10-1', 'YYYY-MM-DD'), '10:00 AM');
INSERT INTO Appointments (appointment_id, doctor_id, patient_id, appointment_date, appointment_time)
VALUES (2, 2, 2, TO_DATE('2024-10-4', 'YYYY-MM-DD'), '2:00 PM');
INSERT INTO Appointments (appointment_id, doctor_id, patient_id, appointment_date, appointment_time)
VALUES (3, 3, 3, TO_DATE('2024-10-6', 'YYYY-MM-DD'), '1:00 PM');
INSERT INTO Appointments (appointment_id, doctor_id, patient_id, appointment_date, appointment_time)
VALUES (5, 5, 5, TO_DATE('2024-10-12', 'YYYY-MM-DD'), '4:00 PM');
```



APPOINTMENT_ID	DOCTOR_ID	PATIENT_ID	APPOINTMENT_DATE	APPOINTMENT_TIME
1	1	1	01-OCT-24	10:00 AM
2	2	2	04-OCT-24	2:00 PM
3	3	3	06-OCT-24	1:00 PM
5	5	5	12-OCT-24	4:00 PM

Inserting data into Medicines Table

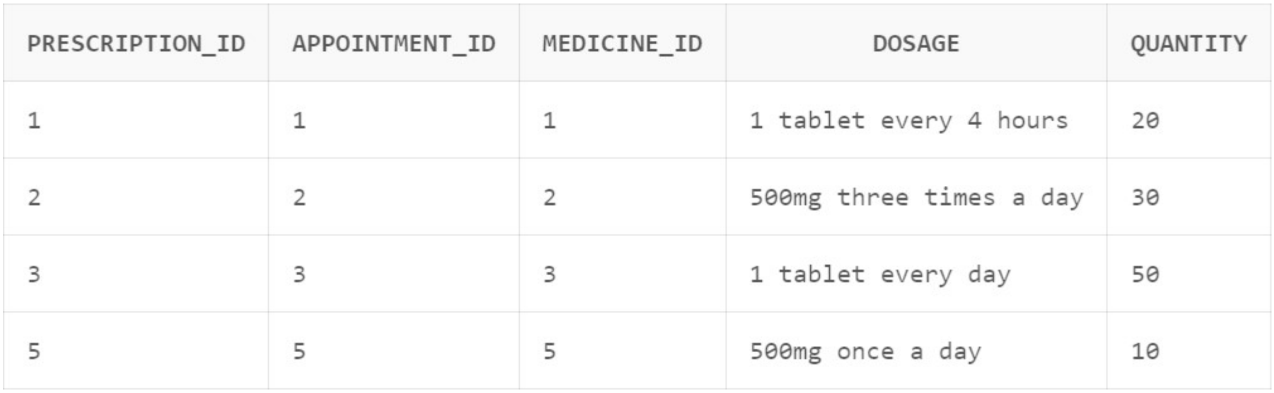
```
INSERT INTO Medicines (medicine_id, name, description, price) VALUES (1, 'Tylenol', 'Pain reliever', 5.99);
INSERT INTO Medicines (medicine_id, name, description, price) VALUES (2, 'Amoxicillin', 'Antibiotic',12.99);
INSERT INTO Medicines (medicine_id, name, description, price) VALUES (3, 'Paracetamol', 'Pain reliever',
7.59);
INSERT INTO Medicines (medicine_id, name, description, price) VALUES (4, 'Dolo-560', 'Antibiotic',4.95);
INSERT INTO Medicines (medicine_id, name, description, price) VALUES (5, 'Ofloxin', 'Pain reliever', 5.87);
INSERT INTO Medicines (medicine_id, name, description, price) VALUES (6, 'Lisinopril','hypertension',12.99);
```



MEDICINE_ID	NAME	DESCRIPTION	PRICE
1	Tylenol	Pain reliever	5.99
2	Amoxicillin	Antibiotic	12.99
3	Paracetamol	Pain reliever	7.59
4	Dolo-560	Antibiotic	4.95
5	Ofloxin	Pain reliever	5.87
6	Lisinopril	hypertension	12.99

Inserting data into Presciptions Table

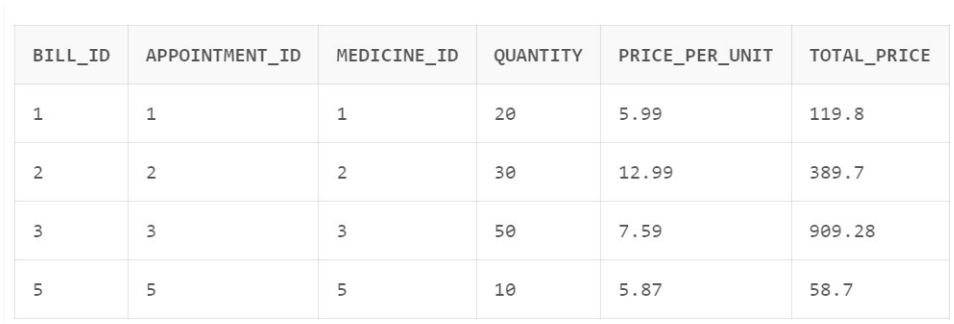
```
INSERT INTO Prescriptions (prescription_id, appointment_id, medicine_id, dosage, quantity) VALUES (1, 1, 1,
'1 tablet every 4 hours', 20);
INSERT INTO Prescriptions (prescription_id, appointment_id, medicine_id, dosage, quantity) VALUES (2,2, 2,
'500mg three times a day', 30);
INSERT INTO Prescriptions (prescription_id, appointment_id, medicine_id, dosage, quantity) VALUES (3, 3, 3,
'1 tablet every day', 50);
INSERT INTO Prescriptions (prescription_id, appointment_id, medicine_id, dosage, quantity) VALUES (5,5, 5,
'500mg once a day', 10);
```



PRESCRIPTION_ID	APPOINTMENT_ID	MEDICINE_ID	DOSAGE	QUANTITY
1	1	1	1 tablet every 4 hours	20
2	2	2	500mg three times a day	30
3	3	3	1 tablet every day	50
5	5	5	500mg once a day	10

Inserting data into Billing Table

INSERT INTO Billing (bill_id, appointment_id, medicine_id, quantity, price_per_unit, total_price) VALUES (1, 1, 1, 20, 5.99, 119.80);
INSERT INTO Billing (bill_id, appointment_id, medicine_id, quantity, price_per_unit, total_price) VALUES (2, 2, 2, 30, 12.99, 389.70);
INSERT INTO Billing (bill_id, appointment_id, medicine_id, quantity, price_per_unit, total_price) VALUES (3, 3, 3, 50, 7.59, 909.282);
INSERT INTO Billing (bill_id, appointment_id, medicine_id, quantity, price_per_unit, total_price) VALUES (5, 5, 5, 10, 5.87, 58.7);



BILL_ID	APPOINTMENT_ID	MEDICINE_ID	QUANTITY	PRICE_PER_UNIT	TOTAL_PRICE
1	1	1	20	5.99	119.8
2	2	2	30	12.99	389.7
3	3	3	50	7.59	909.28
5	5	5	10	5.87	58.7

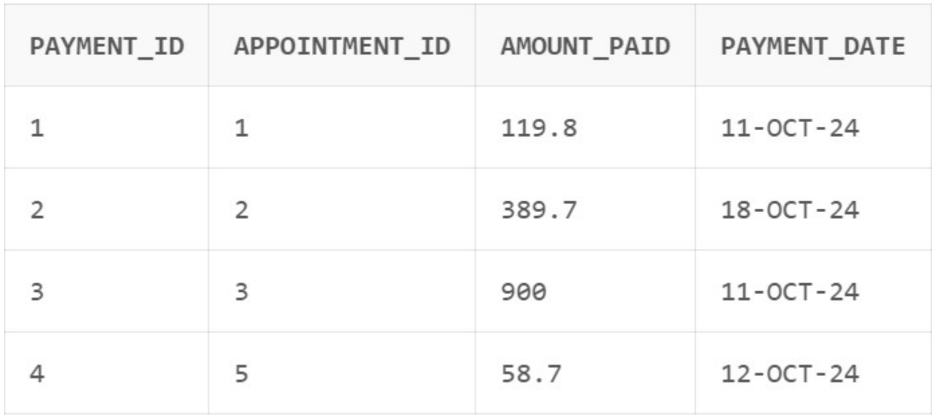
Inserting data into Payments Table

INSERT INTO Payments (payment_id, appointment_id, amount_paid, payment_date) VALUES (1, 1, 119.80, TO_DATE('2024-04-11', 'YYYY-MM-DD'));

INSERT INTO Payments (payment_id, appointment_id, amount_paid, payment_date) VALUES (2, 2, 389.70, TO_DATE('2024-07-18', 'YYYY-MM-DD'));

INSERT INTO Payments (payment_id, appointment_id, amount_paid, payment_date) VALUES (3, 3, 900.00, TO_DATE('2024-03-11', 'YYYY-MM-DD'));

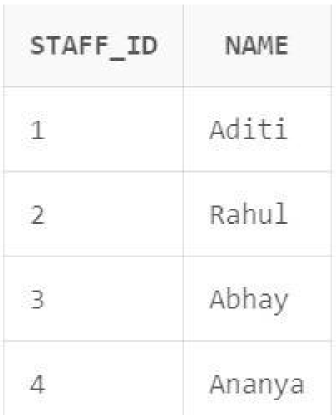
INSERT INTO Payments (payment_id, appointment_id, amount_paid, payment_date) VALUES (4, 5, 58.7, TO_DATE('2024-02-12', 'YYYY-MM-DD'));



PAYMENT_ID	APPOINTMENT_ID	AMOUNT_PAID	PAYMENT_DATE
1	1	119.8	11-OCT-24
2	2	389.7	18-OCT-24
3	3	900	11-OCT-24
4	5	58.7	12-OCT-24

Inserting data into Supporting Staff Table

```
Insert into supporting_staff values(1,'Aditi');
Insert into supporting_staff values(2,'Rahul');
Insert into supporting_staff values(3,'Abhay');
Insert into supporting_staff values(4,'Ananya');
```



STAFF_ID	NAME
1	Aditi
2	Rahul
3	Abhay
4	Ananya

CONCLUSIONS

In conclusion, the doctor-patient management system represents a significant advancement in healthcare technology, offering streamlined management of patient interactions, appointments, prescriptions, and medical records. Its implementation can lead to improved efficiency, accuracy, and communication between healthcare providers and patients, ultimately enhancing the quality of care delivered. By offering customizable solutions and flexible commercial models, such as subscription-based services or one-time purchases, this system can cater to the diverse needs of healthcare organizations and practices. Overall, it stands as a valuable asset in modernizing healthcare delivery and optimizing patient outcomes.

References

- https://drive.google.com/file/d/1DABTfDAGois13Zw9hNT9rwY6HE66V-vu/view?usp=sharing
- https://www.geeksforgeeks.org/how-to-design-a-database-for-healthcare-management-system/
- https://medium.com/@apoorvchowdhry55/hospital-management-system-f21b978a1b8c
- https://chat.openai.com/c/ee4fbf42-3ec8-41c6-b17b-bc29d408c58b
- https://github.com/topics/hospital-management-system