DBMS - Mini Project

HOSPITAL MANAGEMENT SYSTEM

Submitted By:
ANEESH N A
PES1UG20CS050
V Semester Section A

Short Description and Scope of the Project

Hospital Management System is an organised and computerised system designed and programmed to deal with day to day operations and management of the hospital activities. The program can look after inpatients, outpatients, records, database treatments, status illness, billings in the pharmacy and labs. It also maintains hospital information such as room id, doctors in charge. The major problem for the patient nowadays is to get a report after consultation , many hospital managing reports in their system but it's not available to the patient.

1.2 Introduction:

The project Hospital Management system includes registration of patients, storing their details into the system.The software has the facility to give a unique id for every patient and stores the details of every patient and the staff automatically. It includes a search facility to know the current status of each room. Users can search for the availability of a doctor and make appointments.

The Hospital Management System can be entered using a username and password. It is accessible either by an administrator or receptionist. Only they can add data into the database. The data can be retrieved easily. The interface is very user-friendly. The data are well protected for personal use and makes the data processing very fast.

Hospital Management System is powerful, flexible, and easy to use and is designed and developed to deliver real conceivable benefits to hospitals.

Scope of the Project

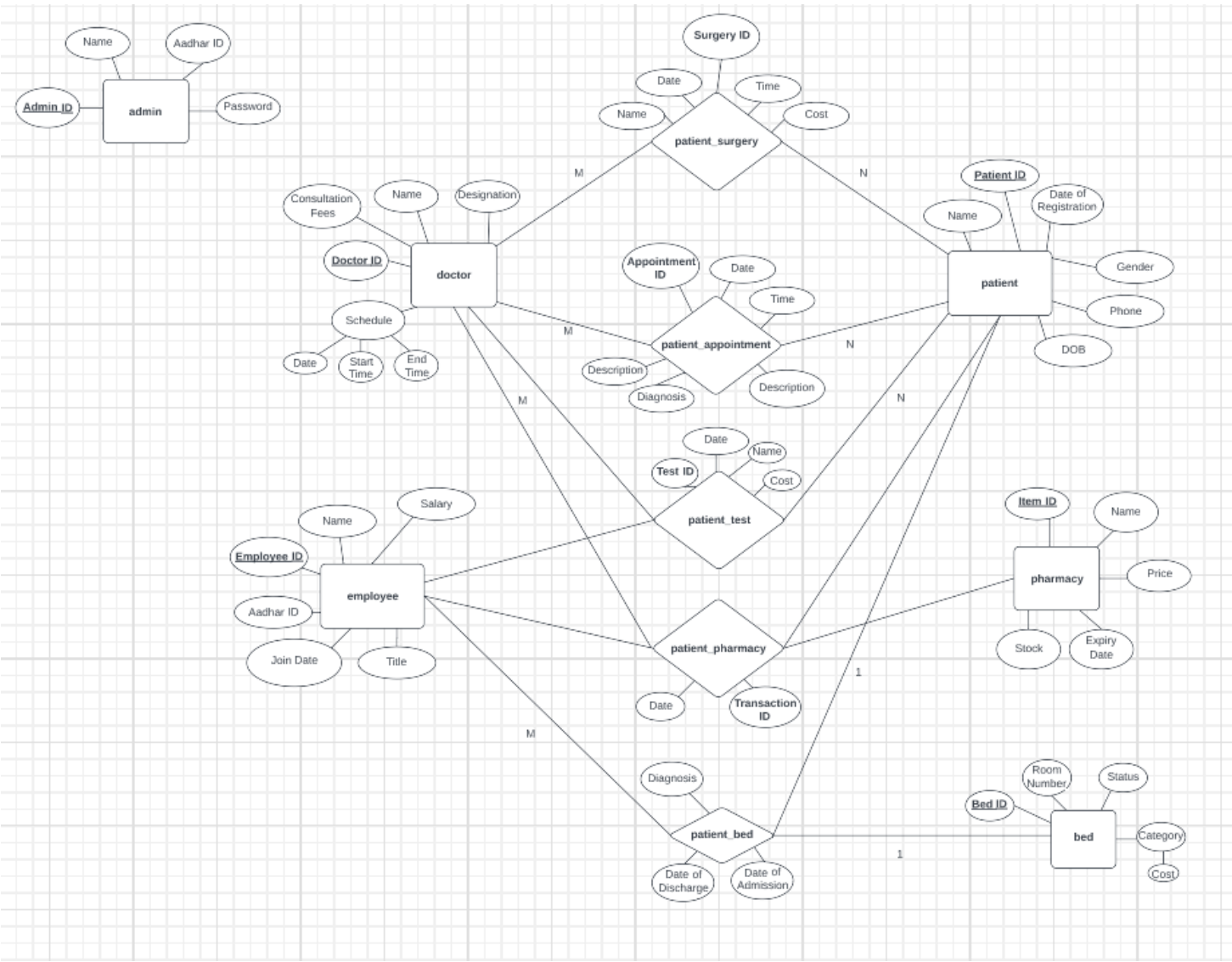
Information about Patients is done by just writing the Patient's name, age and gender.

Diagnosis information to patients is generally recorded on the document, which contains Patient information. It is destroyed after some time period to decrease the paper load in the office.

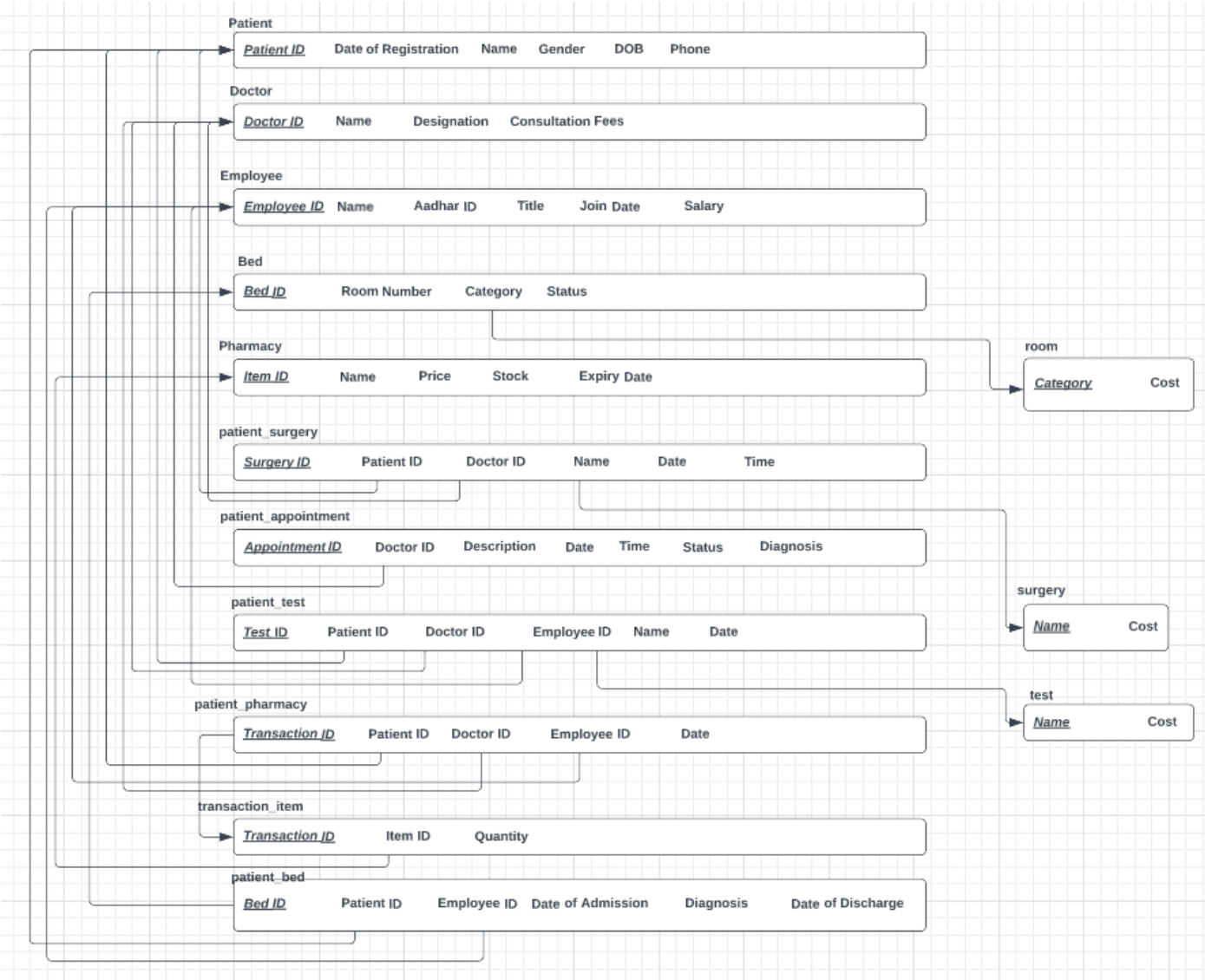
Keep track of the surgeries being done. Keep track of the lab tests, appointments and current availability of the beds and the emergency wards.

Keep track of the inventory of the hospital pharmacy, employee data.

ER Diagram


admin
Name
Aadhar ID
Admin ID
Password
patient_surgery
Surgery ID
Date
Time
Name
Cost
M
N
doctor
Consultation Fees
Name
Designation
Doctor ID
Schedule
Date
Start Time
End Time
patient
Patient ID
Date of Registration
Name
Gender
Phone
DOB
patient_appointment
Appointment ID
Date
Time
Description
Diagnosis
Description
M
N
M
N
patient_test
Date
Name
Test ID
Cost
employee
Name
Salary
Employee ID
Aadhar ID
Join Date
Title
pharmacy
Item ID
Name
Price
Stock
Expiry Date
patient_pharmacy
Date
Transaction ID
1
M
patient_bed
Diagnosis
Date of Discharge
Date of Admission
1
bed
Room Number
Status
Bed ID
Category
Cost

Relational Schema



DDL statements - Building the database

Tables were built using ‘PHPMyAdmin’ Interface.

ALTER TABLE `patient_appointment` ADD CONSTRAINT `A` FOREIGN KEY (`Patient ID`) REFERENCES `patient`(`Patient ID`) ON DELETE RESTRICT ON UPDATE RESTRICT; ALTER TABLE `patient_appointment` ADD CONSTRAINT `B` FOREIGN KEY (`Doctor ID`) REFERENCES `doctor`(`Doctor ID`) ON DELETE RESTRICT ON UPDATE RESTRICT;

ALTER TABLE `patient_bed` ADD CONSTRAINT `C` FOREIGN KEY (`Bed ID`) REFERENCES `bed`(`Bed ID`) ON DELETE RESTRICT ON UPDATE RESTRICT; ALTER TABLE `patient_bed` ADD CONSTRAINT `D` FOREIGN KEY (`Patient ID`) REFERENCES `patient`(`Patient ID`) ON DELETE RESTRICT ON UPDATE RESTRICT;

ALTER TABLE `patient_pharmacy` ADD CONSTRAINT `E` FOREIGN KEY (`Patient ID`) REFERENCES `patient`(`Patient ID`) ON DELETE RESTRICT ON UPDATE RESTRICT; ALTER TABLE `patient_pharmacy` ADD CONSTRAINT` FOREIGN KEY (`Doctor ID`) REFERENCES `doctor`(`Doctor ID`) ON DELETE RESTRICT ON UPDATE RESTRICT; ALTER TABLE `patient_pharmacy` ADD CONSTRAINT `G` FOREIGN KEY (`Employee ID`) REFERENCES `employee`(`Employee ID`) ON DELETE RESTRICT ON UPDATE RESTRICT;

ALTER TABLE `patient_surgery` ADD CONSTRAINT `H` FOREIGN KEY (`Patient ID`) REFERENCES `patient`(`Patient ID`) ON DELETE RESTRICT ON UPDATE RESTRICT; ALTER TABLE `patient_surgery` ADD CONSTRAINT `I` FOREIGN KEY (`Doctor ID`) REFERENCES `doctor`(`Doctor ID`) ON DELETE RESTRICT ON UPDATE RESTRICT; ALTER TABLE `patient_surgery` ADD CONSTRAINT `O` FOREIGN KEY (`Name`) REFERENCES `surgery`(`Name`) ON DELETE RESTRICT ON UPDATE RESTRICT;

ALTER TABLE `patient_test` ADD CONSTRAINT `J` FOREIGN KEY (`Patient ID`) REFERENCES `patient`(`Patient ID`) ON DELETE RESTRICT ON UPDATE RESTRICT; ALTER TABLE `patient_test` ADD CONSTRAINT `K` FOREIGN KEY (`Doctor ID`) REFERENCES `doctor`(`Doctor ID`) ON DELETE RESTRICT ON UPDATE RESTRICT; ALTER TABLE `patient_test` ADD CONSTRAINT `L` FOREIGN KEY (`Employee ID`) REFERENCES `employee`(`Employee ID`) ON DELETE RESTRICT ON UPDATE RESTRICT; ALTER TABLE `patient_test` ADD CONSTRAINT `P` FOREIGN KEY (`Name`) REFERENCES `test`(`Name`) ON DELETE RESTRICT ON UPDATE RESTRICT;

ALTER TABLE `schedule` ADD CONSTRAINT `M` FOREIGN KEY (`Doctor ID`) REFERENCES `doctor`(`Doctor ID`) ON DELETE RESTRICT ON UPDATE RESTRICT;

ALTER TABLE `transaction_item` ADD CONSTRAINT `N` FOREIGN KEY (`Item ID`) REFERENCES `pharmacy`(`Item ID`) ON DELETE RESTRICT ON UPDATE RESTRICT;

Populating the Database

```
INSERT INTO patient VALUES
('PA22467', '2022-01-22','Ashlesha', 'F', '1989-03-02', 3774678352),
('PA22714', '2022-02-02','Atul', 'M', '1947-09-25', 9857146732),
('PA22237', '2022-03-05','Narkhede', 'M', '1979-02-27', 1052377340),
('PA22177', '2022-04-28','Ritu', 'F', '2000-10-17', 1531777997),
('PA22615', '2022-05-24','Yuvraj', 'M', '1955-03-09', 7766156893),
('PA22630', '2022-05-16','Mahajan', 'M', '1993-01-09', 8886308845),
('PA22413', '2022-10-19','Siddhi', 'F', '1973-04-12', 7984131710),
('PA22710', '2022-10-20','Pramod', 'M', '1996-05-10', 6727101903),
('PA22135', '2022-11-13','Patil', 'M', '1987-06-10', 2761351390),
('PA22463', '2022-11-05','Kusum', 'F', '1992-01-16', 3054637657)
('PA22105', '2022-08-14','Abbis', 'M', '1985-08-09', 7766568913),
('PA22330', '2022-11-26','Tiwary', 'M', '2022-11-09', 8883088645),
('PA22193', '2022-01-09','SushantH', 'M', '1973-04-12', 7841391710),
('PA22150', '2022-04-21','Deborah', 'F', '2021-05-10', 6771012903),
('PA22395', '2022-10-13','Kenny', 'F', '1993-08-10', 2761513930),
('PA22653', '2022-10-15','Sebastian', 'M', '2018-04-16', 0546337657);

INSERT INTO doctor VALUES
('DC115', 'Dr. Dana Lepcha', 'Cardiovascular Surgeon', 1500),
('DC804', 'Dr. Akash Saxena', 'Neurologist', 600),
('DC548', 'Dr. Gnana Haokip', 'Resident', 400),
('DC504', 'Dr. Sudha Bharti', 'Paediatric Surgeon', 1100),
('DC110', 'Dr. Meena Sharma', 'Opthalmologist', 800),
('DC340', 'Dr. Rupak Das', 'General Physician', 400),
('DC351', 'Dr. Byomkesh Dutt', 'Resident', 400),
('DC125', 'Dr. Tara Sarkar', 'Dermatologist', 500),
('DC009', 'Dr. Bimal Dubey', 'General Surgeon', 1000),
('DC132', 'Dr. Kumar Dey', 'Paediatrician Attending', 400),
('DC556', 'Dr. David Hume', 'Neurologist', 600),
('DC248', 'Dr. Ajay Yadav', 'Gynaecologist Attending', 600),
('DC031', 'Dr. BL Sharma', 'Neuro Surgeon', 1000),
('DC165', 'Dr. TP Yadav', 'ENT Surgeon', 800),
('DC850', 'Dr. Neha Gupta', 'Oncologist', 1000),
('DC888', ' 1000Dr. Virat Bhatt', 'Nephrologist Attending', 700);

INSERT INTO employee VALUES
('EMP815', 'Pawan', 872281801815, 'Accountant', '2020-02-06', 50000),
('EMP306', 'Hiray', 145230558804, 'Manager', '2022-04-07', 40000),
('EMP837', 'Anand', 609283275848, 'Lab Technician', '2022-05-31', 20000),
('EMP265', 'Nitin', 632226495804, 'Lab Technician', '2020-03-11', 15000),
('EMP508', 'Shirole', 882250561810, 'Lab Technician', '2022-01-18', 30000),
('EMP691', 'Alia', 884269593840, 'Pharmacist', '2020-07-17', 20000),
('EMP451', 'Bhatt', 144245673851, 'Pharmacist', '2020-08-11', 25000),
('EMP794', 'Shahrukh', 978279201825, 'Receptionist', '2020-05-21', 35000),
('EMP082', 'Gautam', 429208230809, 'Receptionist', '2020-07-01', 30000),
('EMP789', 'Shankar', 035278891832, 'Accountant', '2022-06-17', 45000),
```

```
('EMP557', 'Aruna', 633255159840, 'Nurse', '2022-10-18', 20000),
('EMP246', 'Deepthi', 819224692843, 'Nurse', '2022-01-20', 25000),
('EMP038', 'Gowri', 107203555814, 'Nurse', '2020-09-22', 10000),
('EMP164', 'Kumar', 527216625867, 'Nurse', '2021-10-26', 25000),
('EMP857', 'Amitha', 054285013873, 'Nurse', '2021-12-20', 20000);


INSERT INTO bed VALUES
(1 , '001', 'General Ward', 'Vacant'),
(2 , '001', 'General Ward', 'Vacant'),
(3 , '001', 'General Ward', 'Vacant'),
(4 , '001', 'General Ward', 'Occupied'),
(5 , '001', 'General Ward', 'Occupied'),
(6 , '002', 'Emergency Ward', 'Vacant'),
(7 , '002', 'Emergency Ward', 'Vacant'),
(8 , '002', 'Emergency Ward', 'Vacant'),
(9 , '002', 'Emergency Ward', 'Occupied'),
(10, '002', 'Emergency Ward', 'Occupied'),
(11, '003', 'Deluxe', 'Vacant'),
(12, '004', 'Executive Deluxe', 'Vacant'),
(13, '005', 'Twin Sharing', 'Vacant'),
(14, '005', 'Twin Sharing', 'Vacant'),
(15, '006', 'ICU', 'Occupied'),
(16, '007', 'ICU', 'Vacant'),
(17, '101', 'General Ward', 'Vacant'),
(18, '101', 'General Ward', 'Vacant'),
(19, '101', 'General Ward', 'Vacant'),
(20, '101', 'General Ward', 'Vacant'),
(21, '102', 'Deluxe', 'Vacant'),
(22, '103', 'Executive Deluxe', 'Vacant'),
(23, '104', 'NICU', 'Vacant'),
(24, '105', 'NICU', 'Occupied'),
(25, '106', 'ICU', 'Occupied');


INSERT INTO room cost
('General Ward', 500),
('Emergency Ward', 800),
('Deluxe', 1000),
('Executive Deluxe', 2000),
('Twin Sharing', 800),
('ICU', 2500),
('NICU', 3000);
```

```
INSERT INTO test VALUES
('ECG', 500),
('EKG', 1000),
('Lipid Panel', 300),
('HbA1c', 500),
('MRI HR', 10000),
('CT HR', 5000),
('Ultrasound', 2200),
('Dialysis', 2000),
('Blood Panel', 1500),
('Thyroid Panel', 300),
('ECHO', 2500),
('CRP', 1200);


INSERT INTO surgery VALUES
('C-section', 100000 ),
('Joint replacement', 200000),
('Full hysterectomty', 50000),
('Heart valve replacement', 400000),
('Appendectomy', 50000),
('Carotid endarterectomy', 20000),
('Cataract surgery', 40000),
('Cholecystectomy', 60000),
('Coronary artery bypass', 400000)
('Tymphanoplasty', 50000)
('Aneurysm surgery', 30000);


INSERT INTO pharmacy VALUES
('MED171', 'Cyproheptadine', 105.5, 10, '2022-11-02'),
('MED246', 'Diphenhydramine', 100.0, 108, '2025-02-22'),
('MED323', 'Promethazine', 150.5, 104, '2025-03-30'),
('MED417', 'Pheniramine', 20.0, 120, '2022-11-15'),
('MED561', 'Chlorpheniramine', 20.0, 140, '2022-12-28'),
('MED663', 'Pyrilamine', 30.0, 10, '2024-06-08'),
('MED668', 'Pyrilamine', 35.0, 25, '2025-10-08'),
('MED741', 'Epinephrine', 10.0, 130, '2025-07-13'),
('MED871', 'Atropine', 235.6, 10, '2023-02-09'),
('MED842', 'Atropine', 200.0, 30, '2024-12-19'),
('MED913', 'Isoproterenol', 105.5, 10, '2025-08-10'),
('MED146', 'Paracetamol', 10.0, 130, '2025-08-12'),
('MED119', 'Paracetamol', 12.0, 100, '2024-01-01'),
('MED108', 'Diclofenac', 50.0, 100, '2023-10-29'),
('MED105', 'Tylenol', 180.0, 10, '2022-12-02');
```

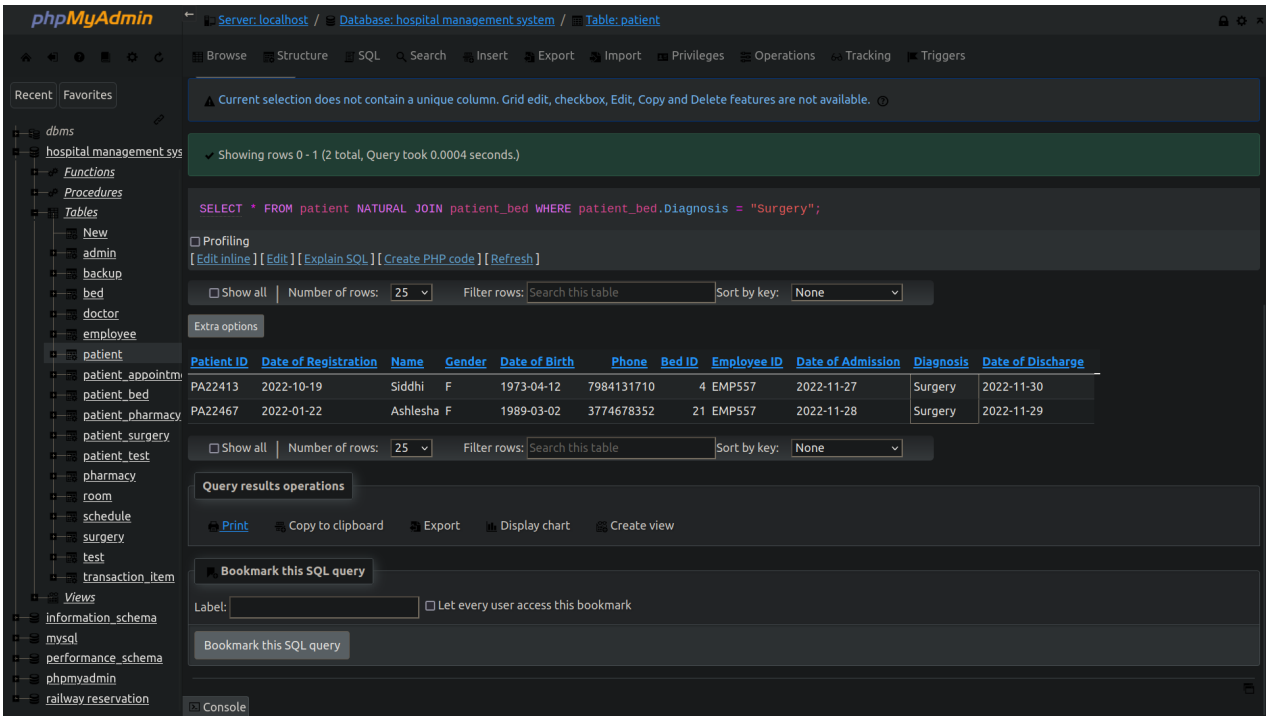
```
INSERT INTO schedule VALUES
('DC804', '2022-11-28', '00:00', '00:00'),
('DC340', '2022-11-29', '11:00', '13:30'),
('DC850', '2022-11-29', '17:30', '18:00'),
('DC165', '2022-11-30', '18:00', '18:30'),
('DC115', '2022-11-30', '10:00', '11:00'),
('DC115', '2022-11-30', '12:30', '13:30'),
('DC125', '2022-12-01', '00:00', '00:00'),
('DC888', '2022-12-02', '10:30', '11:30'),
('DC888', '2022-12-03', '13:00', '13:30'),
('DC031', '2022-12-03', '00:00', '00:00');

INSERT INTO patient_appointment VALUES
('AP001', 'PA22615', 'DC556', 'Poor Grip and Balance', '2022-11-20', '11:00', 'Booked',
'Weakness of Nerves'),
('AP002', 'PA22710', 'DC850', 'Sharp Headache and Mass', '2022-11-21', '18:00', 'Booked',
'CT Scan'),
('AP003', 'PA22467', 'DC248', 'Stomach Ache and Mood Swings', '2022-11-22', '12:30',
'Cancelled', NULL),
('AP004', 'PA22413', 'DC165', 'Ear Pain', '2022-11-23', '11:30', 'Cancelled', NULL),
('AP005', 'PA22467', 'DC248', 'Stomach Ache and Mood Swings', '2022-11-24', '18:30',
'Booked', 'Ultrasound Scanning'),
('AP006', 'PA22135', 'DC009', 'Mass over the joint', '2022-11-25', '19:00', 'Booked', 'MRI
Scan'),
('AP007', 'PA22714', 'DC115', 'Chest Pain', '2022-11-26', '18:00', 'Booked', 'ECG and ECHO
Test'),
('AP008', 'PA22630', 'DC125', 'Rash', '2022-11-26', '13:30', 'Cancelled', NULL),
('AP009', 'PA22413', 'DC165', 'Ear Pain and Dizziness', '2022-11-27', '12:00', 'Booked',
NULL),
('AP010', 'PA22710', 'DC850', 'Post Scan', '2022-11-27', '18:30', 'Booked', NULL),
('AP011', 'PA22237', 'DC110', 'Blurred Vision', '2022-11-27', '18:30', 'Cancelled', NULL),
('AP012', 'PA22467', 'DC248', 'Post Scan', '2022-11-28', '13:00', 'Booked', NULL),
('AP013', 'PA22413', 'DC248', 'PCOD', '2022-11-28', '13:30', 'Booked', NULL),
('AP014', 'PA22714', 'DC115', 'Post Scan', '2022-11-29', '11:00', 'Booked', NULL),
('AP015', 'PA22135', 'DC009', 'Post Scan', '2022-11-29', '17:30', 'Booked', NULL);

INSERT INTO patient_test VALUES
('TE001', 'PA22615', 'DC556', 'EMP265', 'EKG', '2022-11-21'),
('TE002', 'PA22710', 'DC850', 'EMP837', 'CT Scan', '2022-11-23'),
('TE003', 'PA22467', 'DC248', 'EMP265', 'Ultrasound', '2022-11-25'),
('TE004', 'PA22135', 'DC009', 'EMP837', 'MRI HR', '2022-11-25'),
('TE005', 'PA22714', 'DC115', 'EMP837', 'ECG', '2022-11-26'),
('TE006', 'PA22714', 'DC115', 'EMP265', 'ECHO', '2022-11-26');

INSERT INTO patient_pharmacy VALUES
('TRN001', 'PA22615', 'DC556', 'EMP451', '2022-11-20'),
('TRN002', 'PA22710', 'DC850', 'EMP451', '2022-11-21'),
('TRN003', 'PA22615', 'DC556', 'EMP691', '2022-11-22'),
('TRN004', 'PA22467', 'DC248', 'EMP691', '2022-11-24'),
('TRN005', 'PA22135', 'DC009', 'EMP691', '2022-11-25'),
('TRN006', 'PA22467', 'DC248', 'EMP451', '2022-11-26');
```

```
INSERT INTO transaction_item VALUES
('TRN001', 'MED871', 1),
('TRN001', 'MED105', 1),
('TRN002', 'MED119', 10),
('TRN002', 'MED105', 2),
('TRN002', 'MED246', 8),
('TRN003', 'MED663', 2),
('TRN003', 'MED561', 5),
('TRN004', 'MED108', 5),
('TRN004','MED171', 5)
('TRN005', 'MED741', 5),
('TRN006', 'MED871', 1),
('TRN006', 'MED119', 5);

INSERT INTO patient_surgery VALUES
('SUR342', 'PA22714', 'DC115', 'Coronary artery bypass', '2022-11-30', '10:30'),
('SUR889', 'PA22467', 'DC009', 'Cholecystectomy', '2022-11-28', '15:00'),
('SUR990', 'PA22714', 'DC115', 'Heart valve replacement', '2022-12-03', '17:00'),
('SUR889', 'PA22135', 'DC009', 'Joint replacement', '2022-12-01', '19:30'),
('SUR463', 'PA22463', 'DC009', 'C-section', '2022-12-01', '10:00'),
('SUR868', 'PA22615', 'DC031', 'Carotid endarterectomy', '2022-11-21', '05:30'),
('SUR543', 'PA22413', 'DC165', 'Tymphanoplasty', '2022-11-27', '18:30'),
('SUR013', 'PA22615', 'DC031', 'Aneurysm surgery', '2022-11-22', '16:00');

INSERT INTO patient_bed VALUES
(4,'PA22413', 'EMP557', '2022-11-27', 'Surgery', '2022-11-30'),
(5,'PA22615', 'EMP857', '2022-11-26', 'Seizures', '2022-12-01'),
(21,'PA22467', 'EMP557', '2022-11-28', 'Surgery', '2022-11-29'),
(24,'PA22330', 'EMP164', '2022-11-26', 'Weakness', '2022-11-30'),
(9,'PA22193', 'EMP038', '2022-11-27', 'Accident', NULL),
(10,'PA22150', 'EMP164', '2022-11-27', 'Accident', NULL),
(15,'PA22395', 'EMP038', '2022-11-26', 'Cardiac Arrest', '2022-12-01');
```

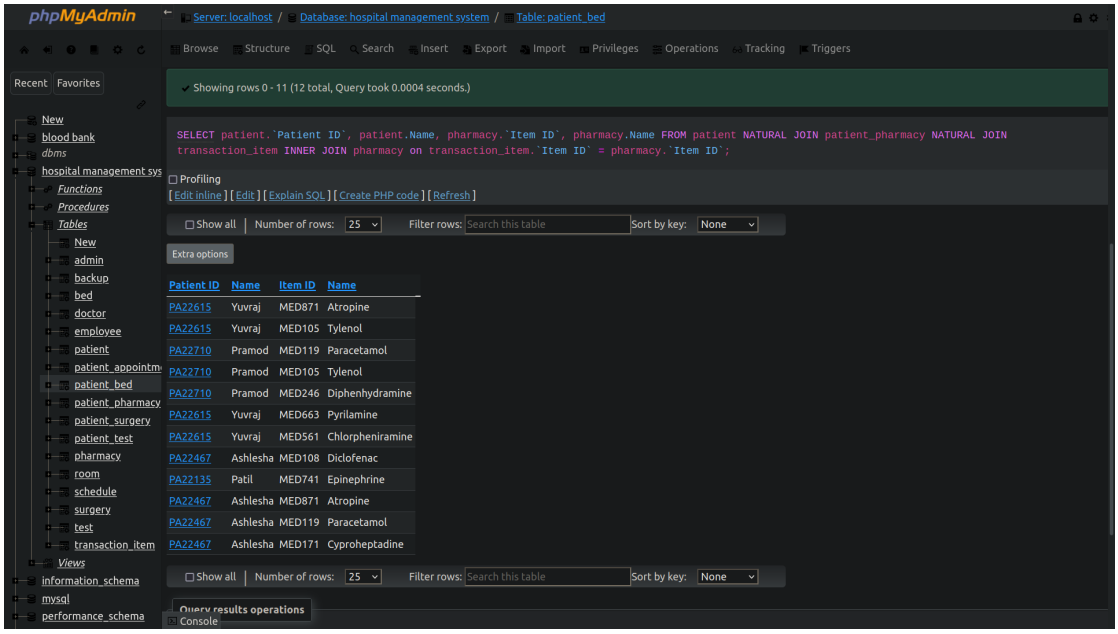
Join Queries

Showcase at least 4 join queries
Write the query in English Language, Show the equivalent SQL statement and also a screenshot of the query and the result

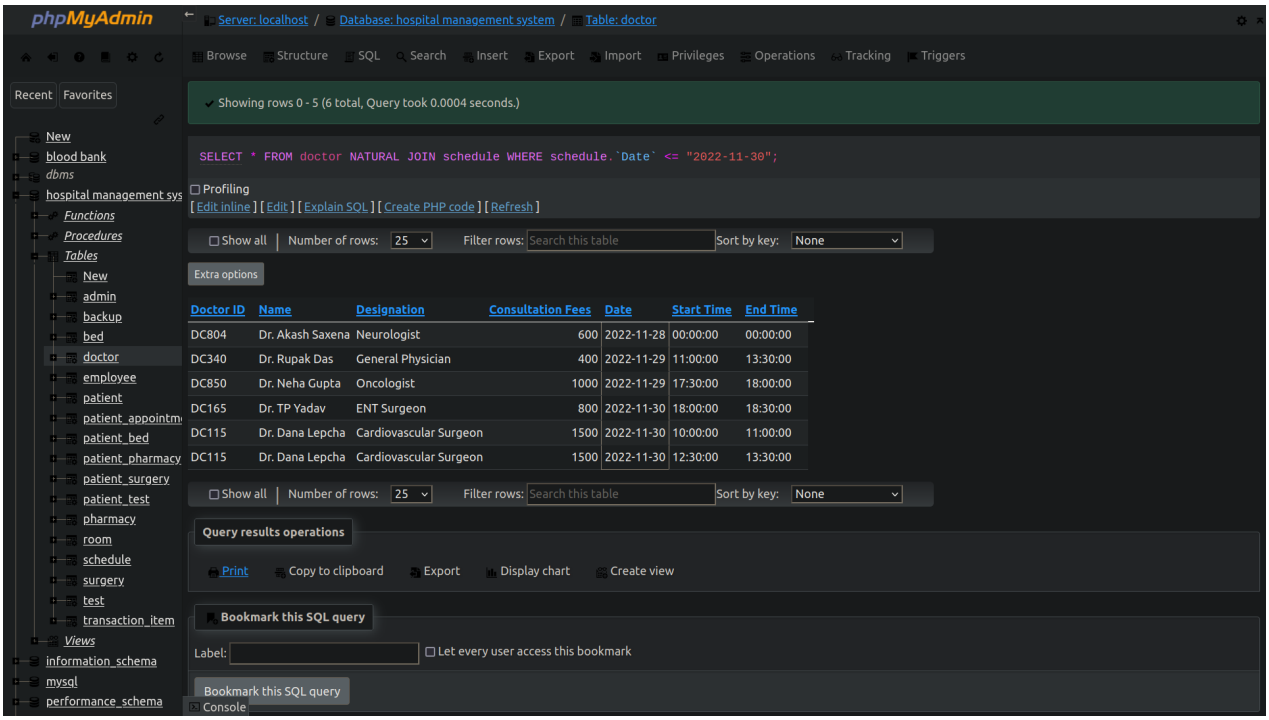
1. Get the list of all the patients who're currently admitted for the surgery.



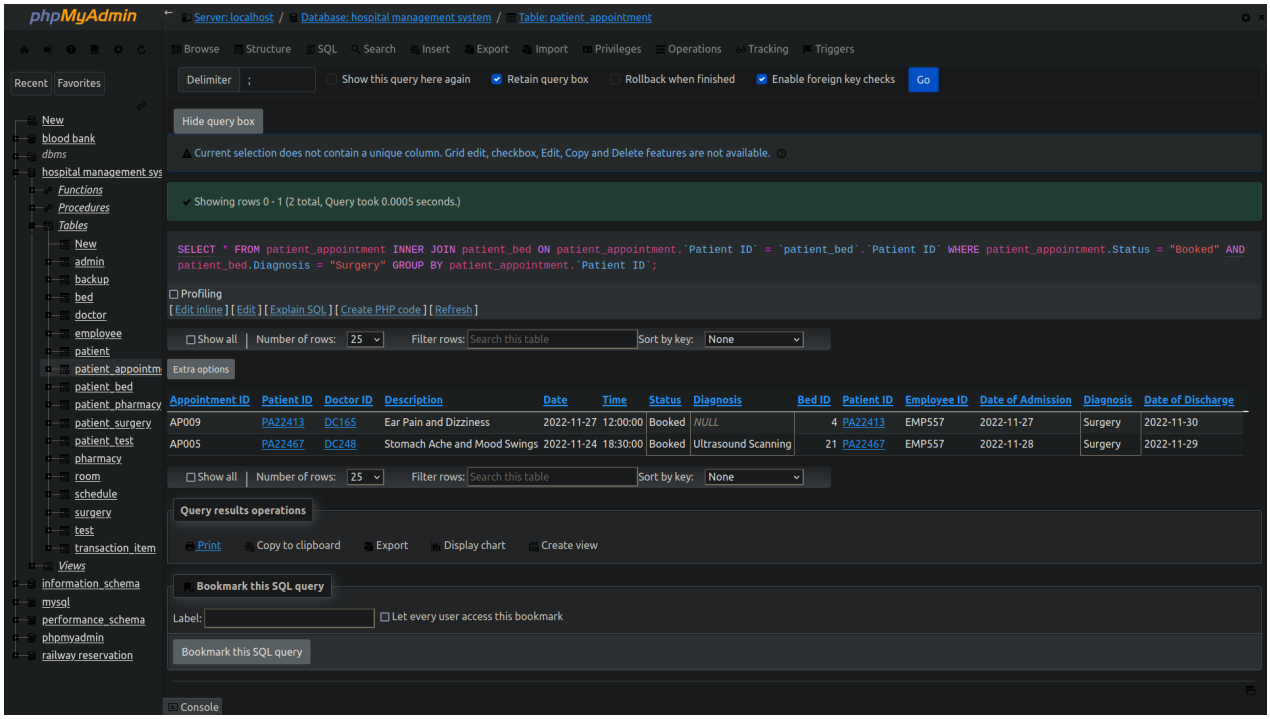
2. Find the list of all the medicines purchased by the patient in the pharmacy for the given time period



3. Find the doctors who wouldn't be available and their unavailability in the schedule within the next three days.



4. Get the list of all the patients who took appointments and then required a surgery for their diagnosis.



JOIN QUERIES

1. SELECT * FROM patient NATURAL JOIN patient_bed WHERE patient_bed.Diagnosis = "Surgery";

2. SELECT patient.`Patient ID`, patient.Name, pharmacy.`Item ID', pharmacy.Name from patient
NATURAL JOIN patient_pharmacy NATURAL JOIN transaction_item INNER JOIN pharmacy on transaction_item.`Item ID` = pharmacy.`Item ID`;

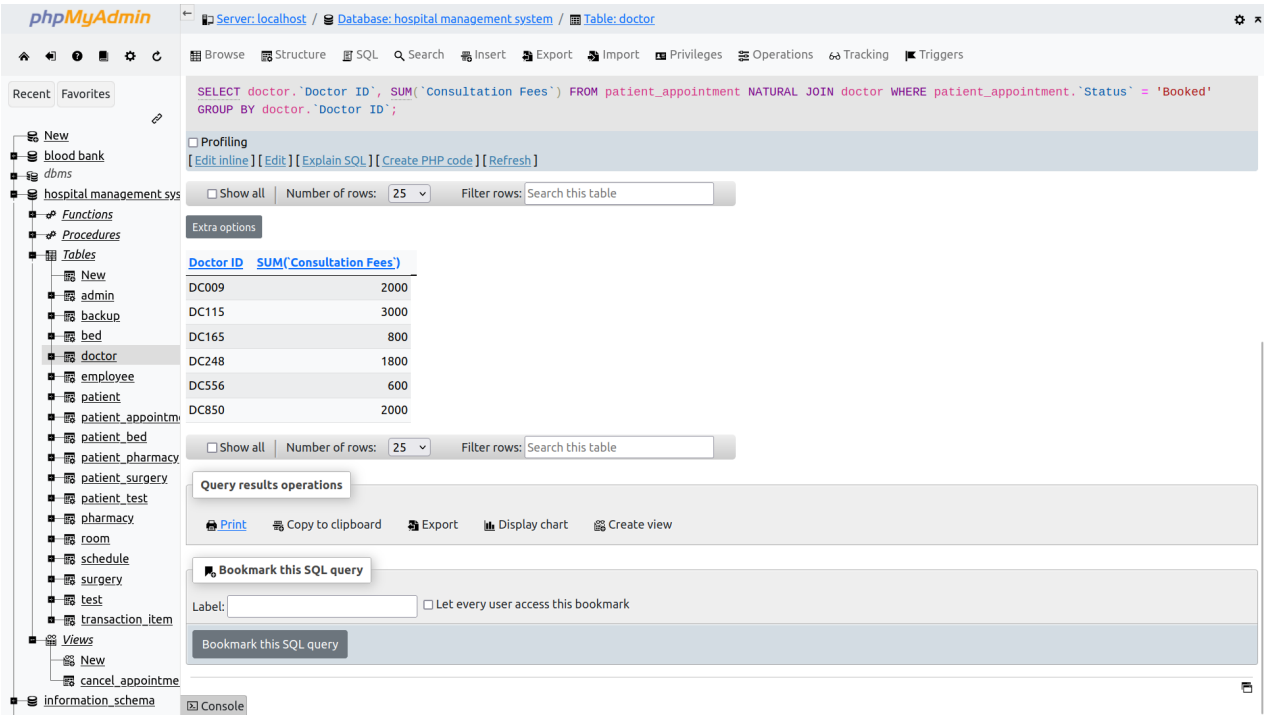
3.
SELECT * FROM doctor NATURAL JOIN schedule
WHERE schedule.`Date` <= "2022-11-30";

4.
SELECT * FROM patient_appointment INNER JOIN patient_bed ON patient_appointment.`Patient ID` = `patient_bed`.`Patient ID`
WHERE patient_appointment.Status = "Booked" AND patient_bed.Diagnosis = "Surgery"
GROUP BY patient_appointment.`Patient ID`;

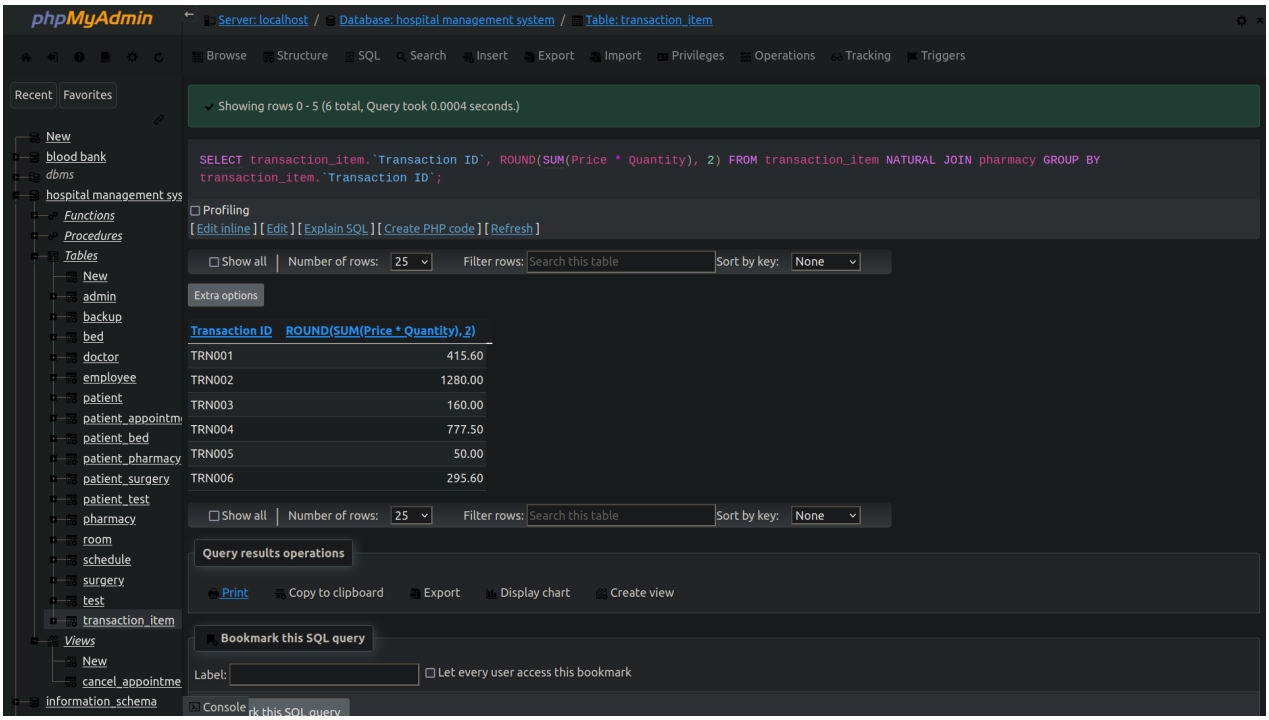
Aggregate Functions

Showcase at least 4 Aggregate function queries
Write the query in English Language, Show the equivalent SQL statement and also a screenshot of the query and the results

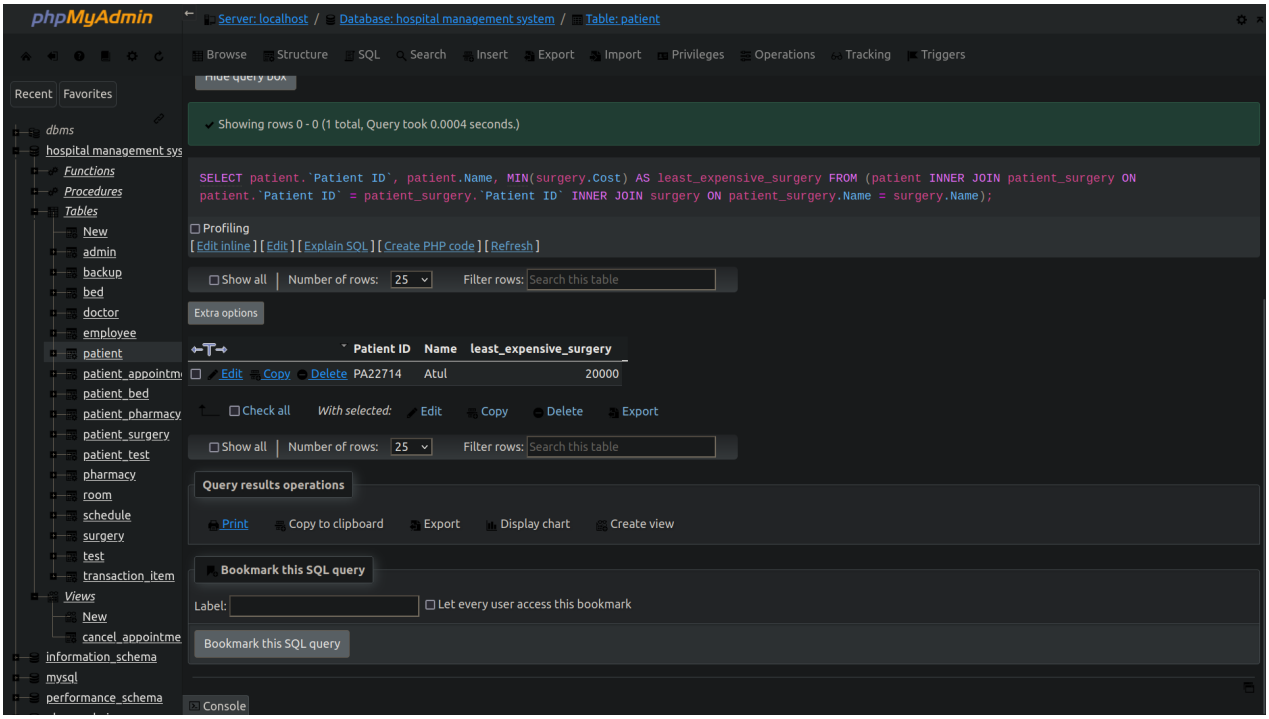
1. List out the total consultation fees earned by the doctors from the OPD over the entire table.



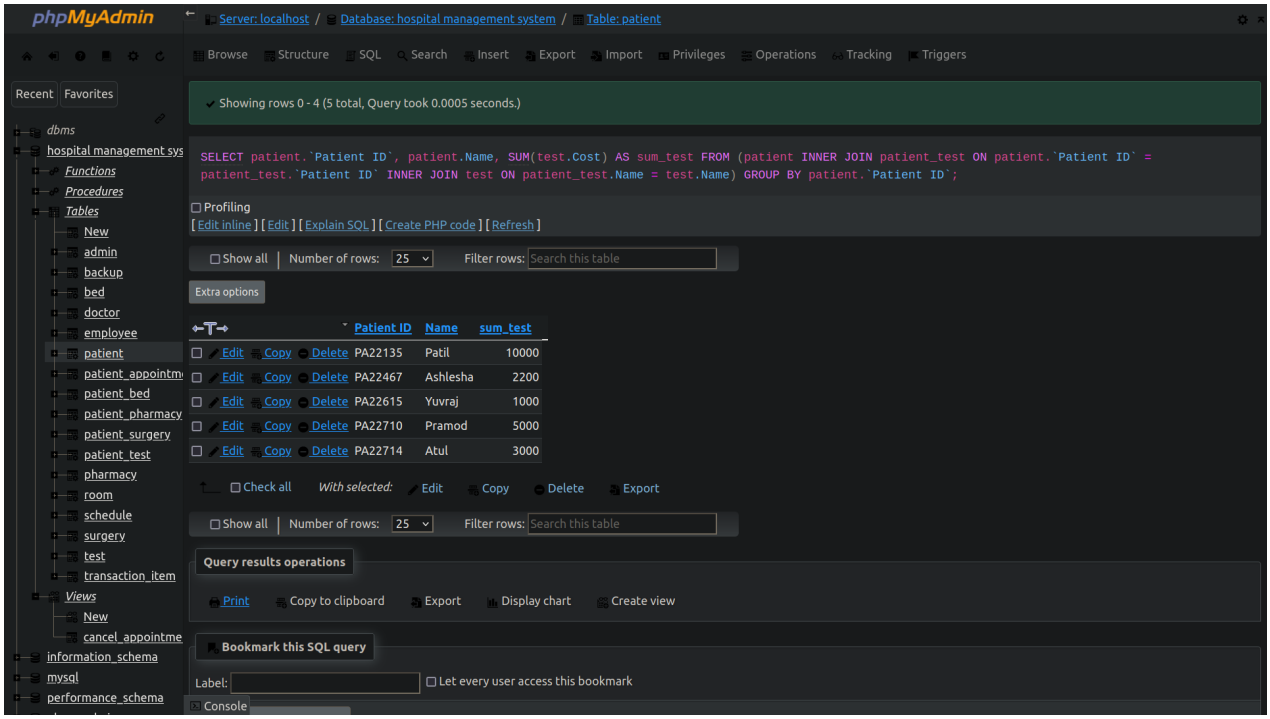
2. Get the total cost of the transactions in the pharmacy.



3. List the patient who had the least expensive surgery among all others.



4. List the total cost incurred on tests by the patients.



AGGREGATE QUERIES

1.
```
SELECT doctor.`Doctor ID`, SUM(`Consultation Fees`) FROM
patient_appointment NATURAL JOIN doctor
WHERE patient_appointment.`Status` = 'Booked'
GROUP BY doctor.`Doctor ID`;
```

2.
```
SELECT transaction_item.`Transaction ID`, ROUND(SUM(Price * Quantity), 2) FROM
transaction_item NATURAL JOIN pharmacy GROUP BY transaction_item.`Transaction ID`;
```

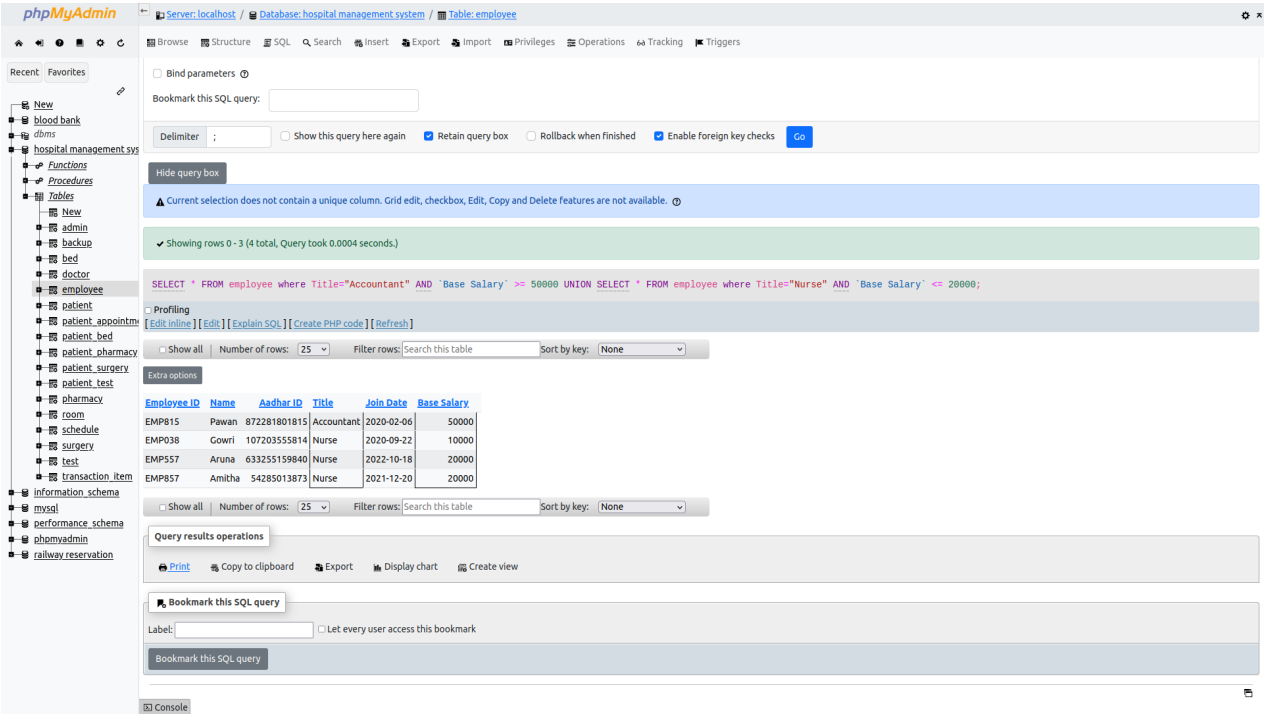
3.
```
SELECT patient.`Patient ID`, patient.Name, MIN(surgery.Cost) AS least_expensive_surgery
FROM
(patient INNER JOIN
patient_surgery ON patient.`Patient ID` = patient_surgery.`Patient ID`
INNER JOIN surgery ON patient_surgery.Name = surgery.Name);
```

4.
```
SELECT patient.`Patient ID`, patient.Name, SUM(test.Cost) AS sum_test FROM (patient INNER
JOIN patient_test ON patient.`Patient ID` = patient_test.`Patient ID`
INNER JOIN test ON patient_test.Name = patient_test.Name) GROUP BY patient.`Patient ID`;
```

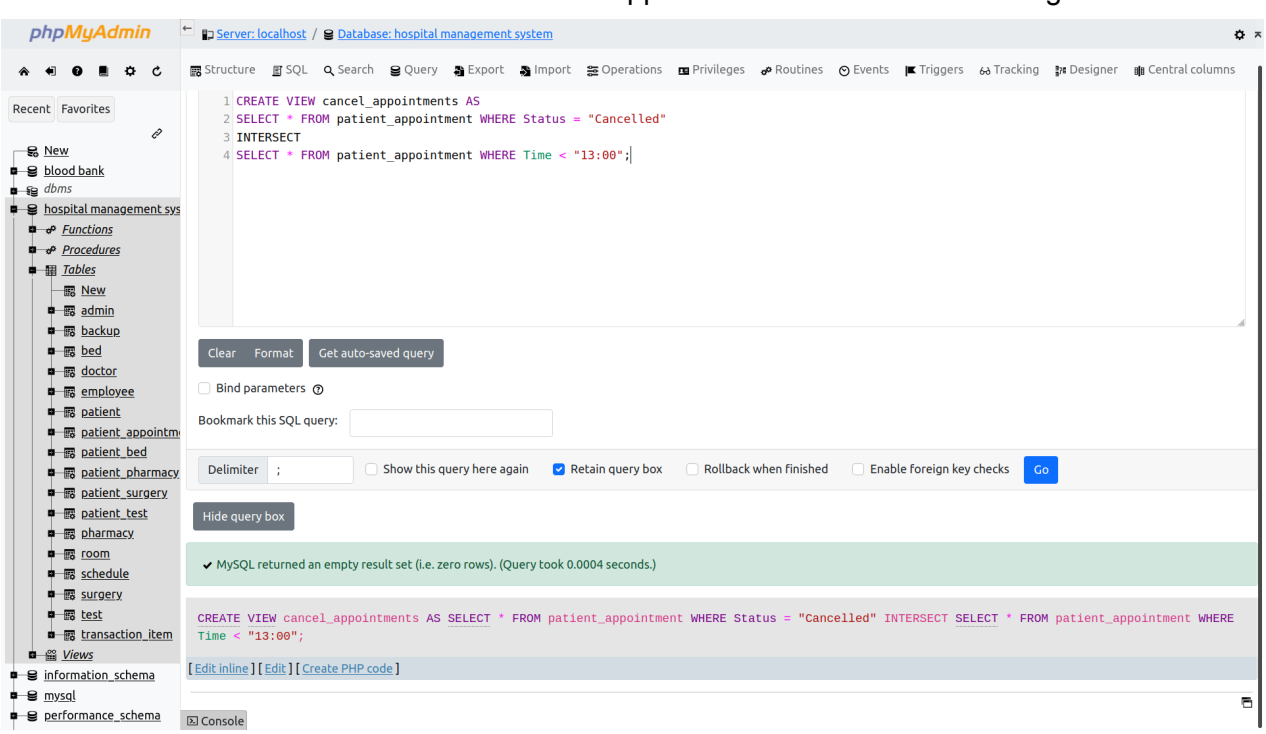
Set Operations

Showcase at least 4 Set Operations queries
Write the query in English Language, Show the equivalent SQL statement and also a screenshot of the query and the results

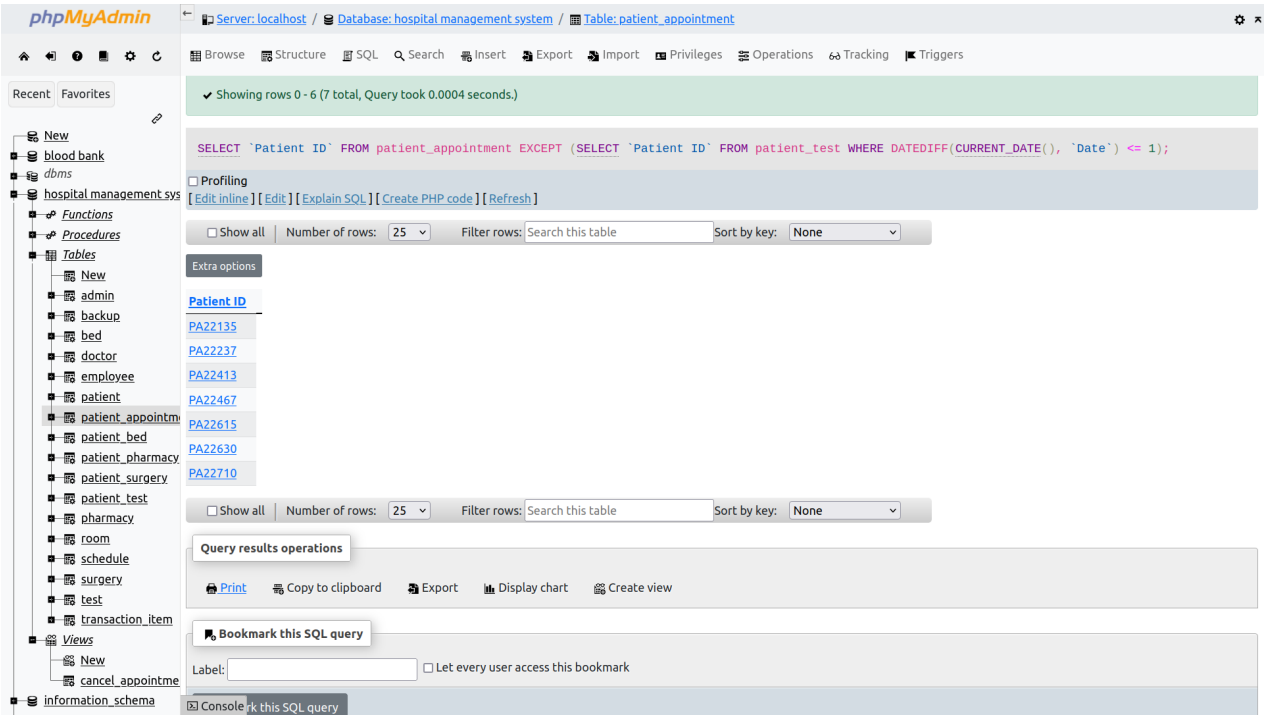
1. List Nurses who are paid <= 20000 and the accountants being paid more than >= 50000



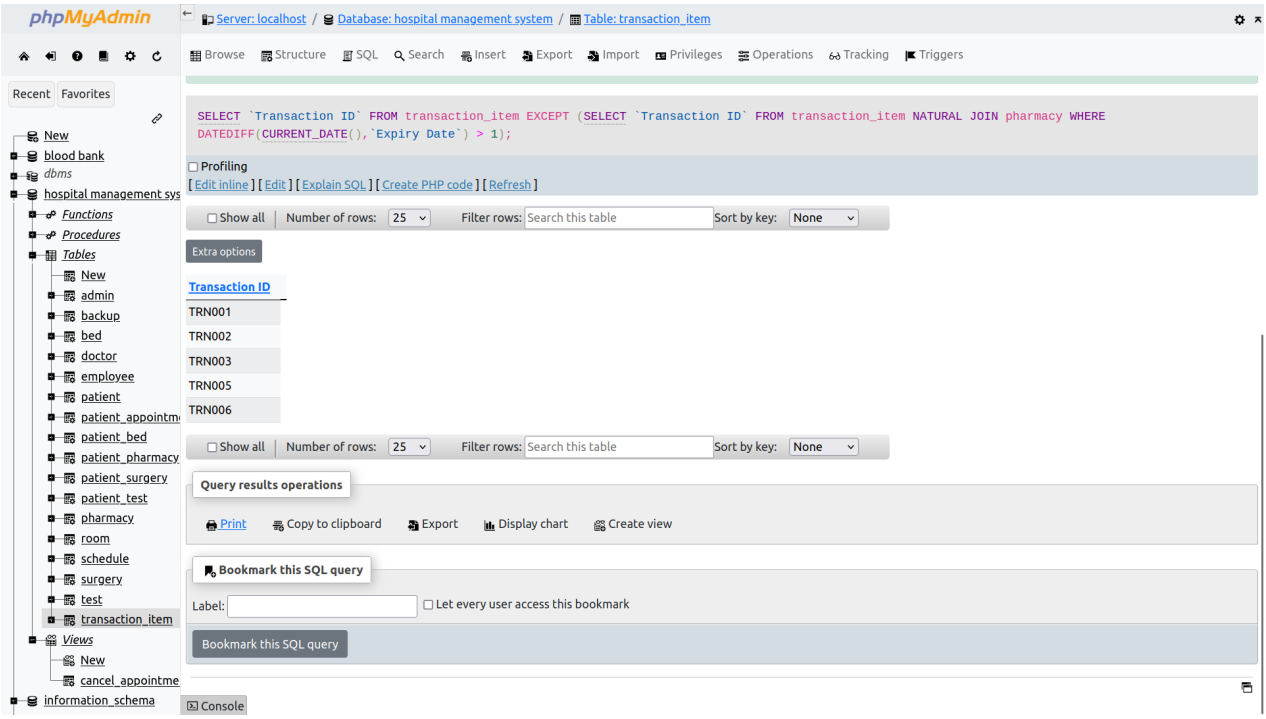
2. Create a view to list out all the cancelled appointments made in the morning.



3. List all the patients who took appointments and got a test done not less than a day ago.



4. LIst out all the transactions where no expired medicines were bought for the patients.



SET

1.
```
SELECT * FROM employee where Title="Accountant" AND Base Salary >= 50000
UNION
SELECT * FROM employee where Title="Nurse" AND Base Salary <= 20000;
```

2.
```
CREATE VIEW cancel_appointments AS
SELECT * FROM patient_appointment WHERE Status = "Cancelled"
INTERSECT
SELECT * FROM patient_appointment WHERE Time < "13:00";
```

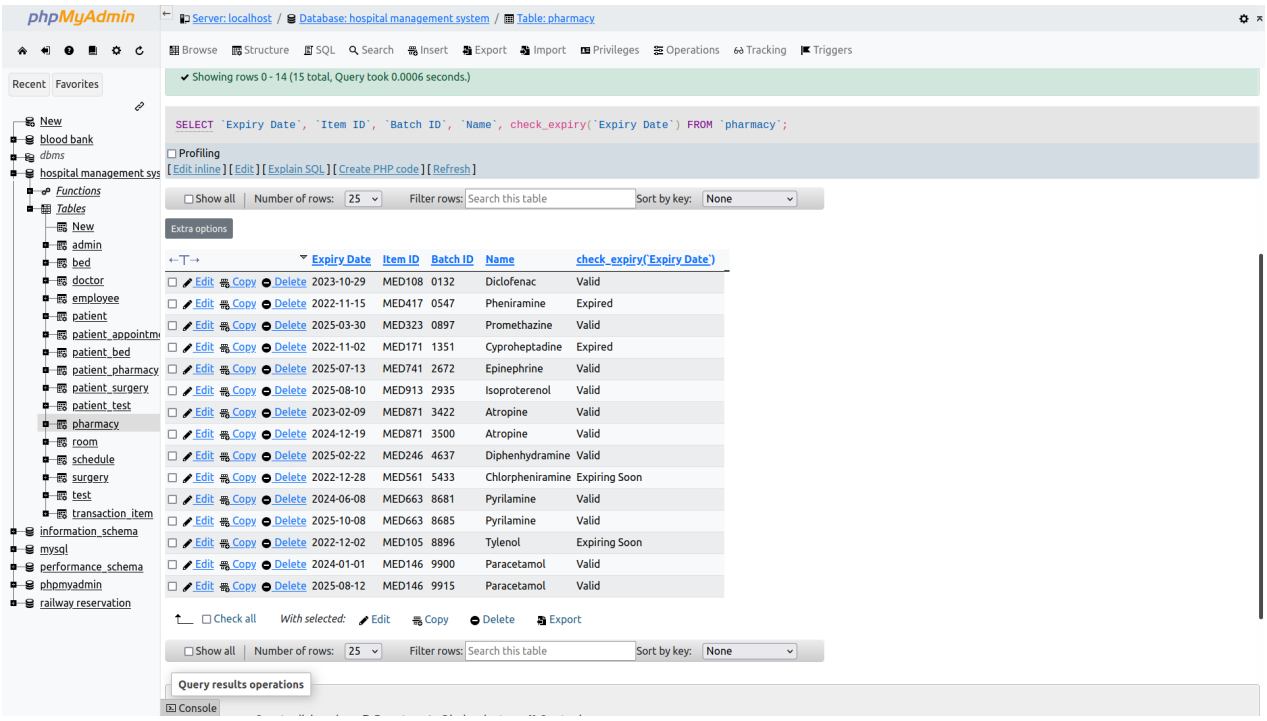
3.
```
SELECT `Patient ID` FROM patient_appointment
EXCEPT
(SELECT `Patient ID` FROM patient_test WHERE DATEDIFF(CURRENT_DATE(), `Date`) <=
1);
```

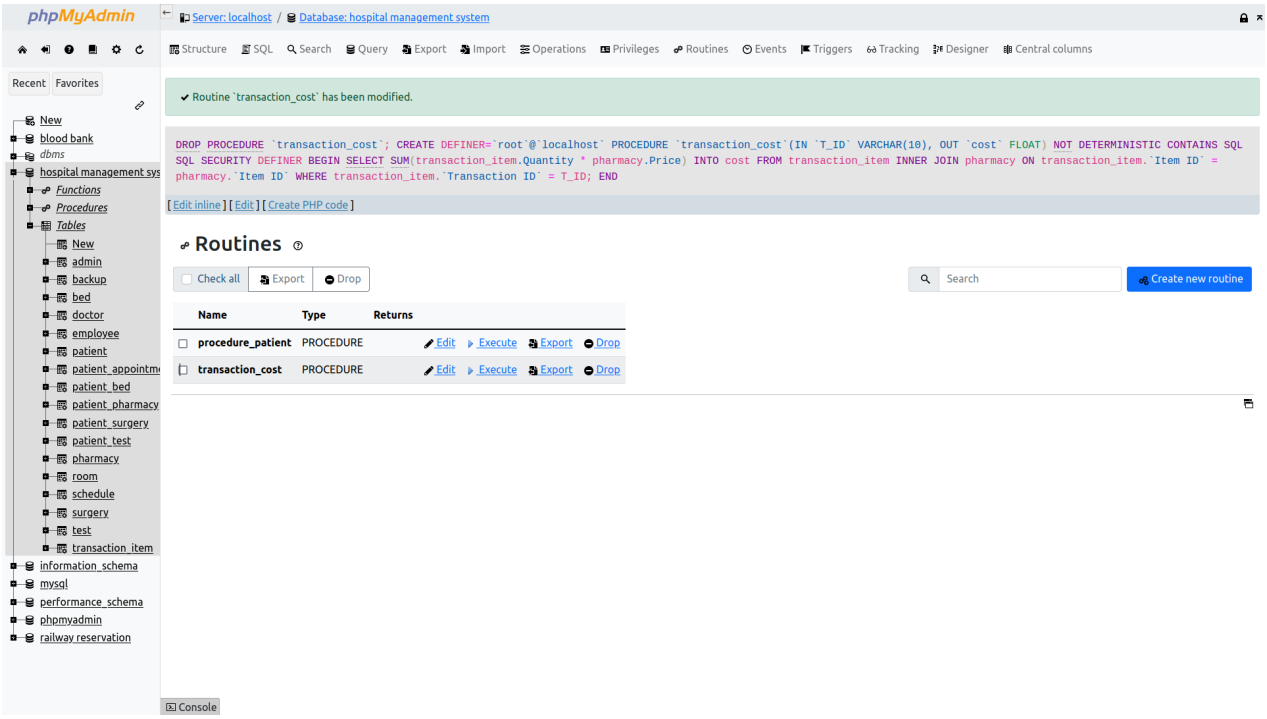
4.
```
SELECT `Transaction ID` FROM transaction_item
EXCEPT
(SELECT `Transaction ID` FROM transaction_item NATURAL JOIN pharmacy
WHERE DATEDIFF(CURRENT_DATE(),`Expiry Date`) > 1);
```

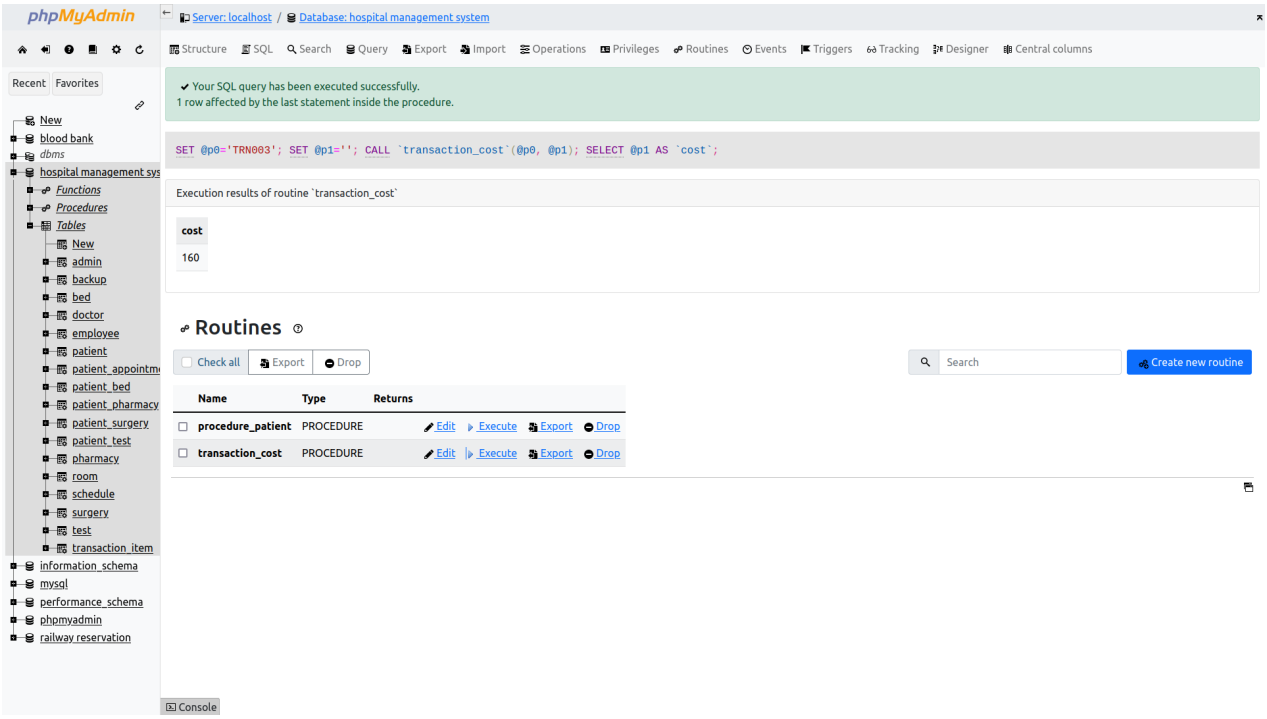
Functions and Procedures

Create a Function and Procedure. State the objective of the function / Procedure. Run and display the results.

```
DELIMITER $$
CREATE FUNCTION check_expiry(D DATE)
RETURNS VARCHAR(20)
BEGIN
DECLARE statement_value VARCHAR(20);
IF D < CURRENT DATE() THEN
SET statement_value = 'Expired';
ELSEIF DATEDIFF(D, CURRENT DATE()) < 180 THEN
SET statement_value = 'Expiring Soon';
ELSE
SET statement_value = 'Valid';
END IF;
RETURN statement_value;
END; $$
```








```
DELIMITER $$
CREATE PROCEDURE transaction_cost(IN T_ID VARCHAR(10),OUT cost FLOAT)
BEGIN
SELECT SUM(transaction_item.Quantity * pharmacy.Price) INTO cost
FROM transaction_item
INNER JOIN pharmacy
ON transaction_item.`Item ID` = pharmacy.`Item ID`
WHERE transaction_item.`Transaction ID` = T_ID;
END;$$
```

TRIGGER

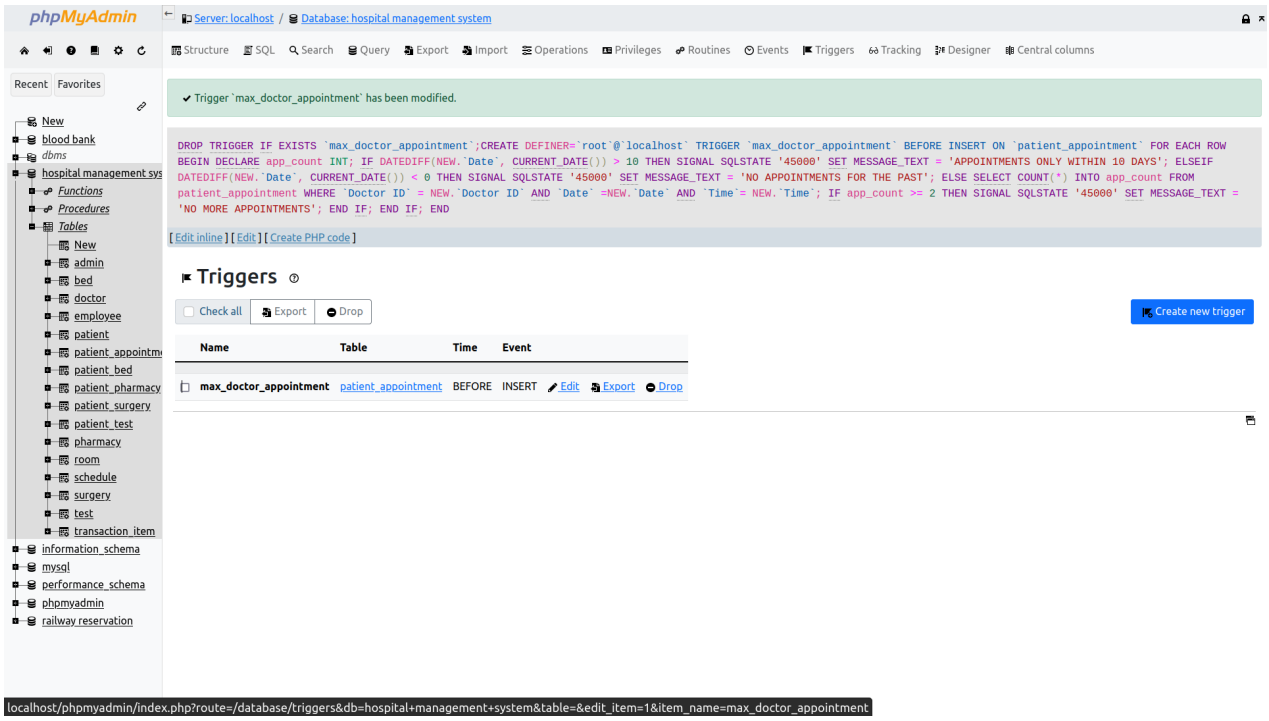
```
DELIMITER $$
BEGIN
DECLARE app_count INT;
IF DATEDIFF(NEW.`Date`, CURRENT_DATE()) > 10 THEN
   SIGNAL SQLSTATE '45000' SET MESSAGE_TEXT = 'APPOINTMENTS ONLY WITHIN 10 DAYS';
ELSEIF DATEDIFF(NEW.`Date`, CURRENT_DATE()) < 0 THEN
   SIGNAL SQLSTATE '45000' SET MESSAGE_TEXT = 'NO APPOINTMENTS FOR THE PAST';
ELSE
        SELECT COUNT(*) INTO app_count
        FROM patient_appointment
        WHERE `Doctor ID` = NEW.`Doctor ID` AND `Date` =NEW.`Date` AND `Time`= NEW.`Time`;

        IF app_count >= 2 THEN
        SIGNAL SQLSTATE '45000' SET MESSAGE_TEXT = 'NO MORE APPOINTMENTS';
        END IF;
END IF;
END;$$
```

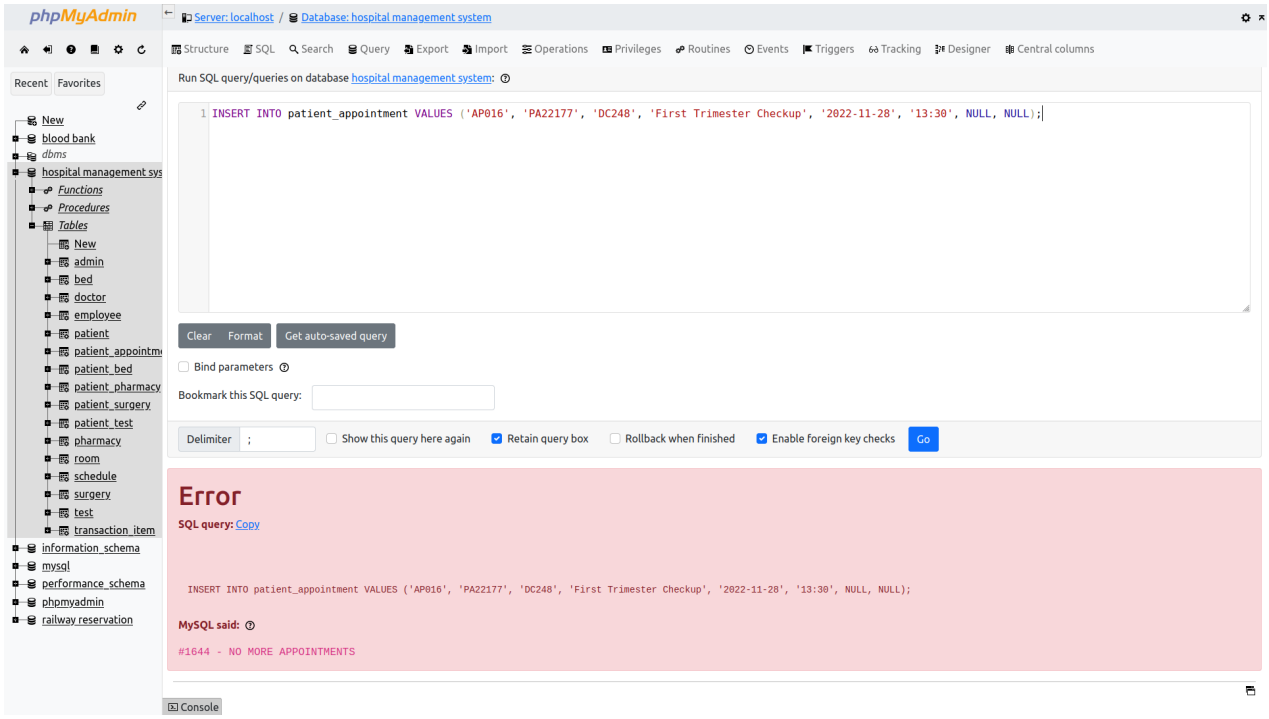
CURSOR

```
BEGIN
DECLARE done INT DEFAULT 0;
DECLARE N VARCHAR(20);
DECLARE I VARCHAR(10);
DECLARE D DATE;
DECLARE cursor_patient CURSOR FOR
SELECT `Patient ID`, `Name`, `Date of Registration` FROM patient;
DECLARE CONTINUE HANDLER FOR NOT FOUND SET done = 1;
OPEN cursor_patient;
label: LOOP
FETCH cursor_patient INTO I, N, D;
IF done = 1 THEN
        LEAVE label;
END IF;
IF DATEDIFF(CURRENT_DATE(), D) < 60 THEN
        INSERT INTO backup VALUES(I, N);
END IF;
END LOOP label;
CLOSE cursor_patient;
END
```

Triggers and Cursors

Create a Trigger and a Cursor. State the objective. Run and display the results.





phpMyAdmin

Server: localhost / Database: hospital management system

Structure | SQL | Search | Query | Export | Import | Operations | Privileges | Routines | Events | Triggers | Tracking | Designer | Central columns

Recent | Favorites

- New
- blood bank
- dbms
- hospital management sys
  - Functions
  - Procedures
  - Tables
    - New
    - admin
    - backup
    - bed
    - doctor
    - employee
    - patient
    - patient_appointm
    - patient_bed
    - patient_pharmacy
    - patient_surgery
    - patient_test
    - pharmacy
    - room
    - schedule
    - surgery
    - test
    - transaction_item
- information_schema
- mysql
- performance_schema
- phpmyadmin
- railway reservation

✔ Routine `procedure_patient` has been modified.

```
DROP PROCEDURE `procedure_patient`; CREATE DEFINER=`root`@`localhost` PROCEDURE `procedure_patient`() NOT DETERMINISTIC CONTAINS SQL SQL SECURITY DEFINER BEGIN DECLARE done INT DEFAULT 0; DECLARE N VARCHAR(20); DECLARE I VARCHAR(10); DECLARE D DATE; DECLARE cursor_patient CURSOR FOR SELECT `Patient ID`, `Name`, `Date of Registration` FROM patient; DECLARE CONTINUE HANDLER FOR NOT FOUND SET done = 1; OPEN cursor_patient; label: LOOP FETCH cursor_patient INTO I, N, D; IF done = 1 THEN LEAVE label; END IF; IF DATEDIFF(CURRENT_DATE(), D) < 60 THEN INSERT INTO backup VALUES(I, N); END IF; END LOOP label; CLOSE cursor_patient; END
```

[Edit inline] [Edit] [Create PHP code]

Routines

Check all | Export | Drop | Search | Create new routine

Name	Type	Returns	
transaction_cost	PROCEDURE		Edit Execute Export Drop
procedure_patient	PROCEDURE		Edit Execute Export Drop

localhost/phpmyadmin/index.php?route=/database/routines&db=hospital+management+system&table=&edit_item=1&item_name=procedure_patient&item_type=PROCEDURE

phpMyAdmin

Server: localhost / Database: hospital management system / Table: backup

Browse | Structure | SQL | Search | Insert | Export | Import | Privileges | Operations | Tracking | Triggers

Recent | Favorites

- New
- blood bank
- dbms
- hospital management sys
  - Functions
  - Procedures
  - Tables
    - New
    - admin
    - backup
    - bed
    - doctor
    - employee
    - patient
    - patient_appointm
    - patient_bed
    - patient_pharmacy
    - patient_surgery
    - patient_test
    - pharmacy
    - room
    - schedule
    - surgery
    - test
    - transaction_item
- information_schema
- mysql
- performance_schema
- phpmyadmin
- railway reservation

✔ Showing rows 0 - 6 (7 total, Query took 0.0002 seconds.)

```
SELECT * FROM `backup`
```

Profiling

[Edit inline] [Edit] [Explain SQL] [Create PHP code] [Refresh]

Show all | Number of rows: 25 | Filter rows: Search this table

Extra options

ID	Name
PA22135	Patil
PA22330	Tiwary
PA22395	Kenny
PA22413	Siddhi
PA22463	Kusum
PA22653	Sebastian
PA22710	Pramod

Show all | Number of rows: 25 | Filter rows: Search this table

Query results operations

Print | Copy to clipboard | Export | Display chart | Create view

Bookmark this SQL query

Label: | Let every user access this bookmark

Bookmark this SQL query

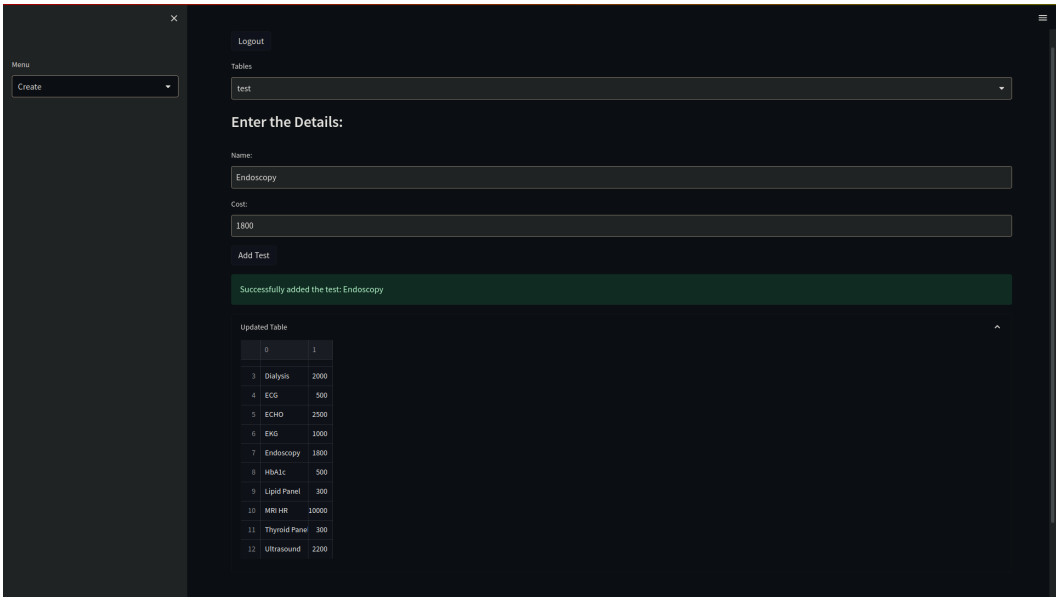
Console

Developing a Frontend

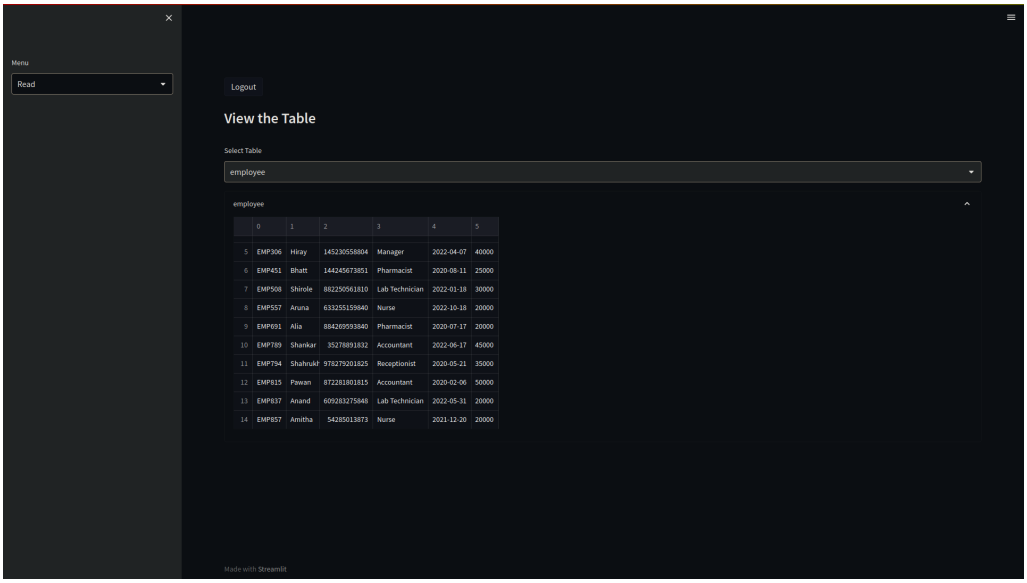
The frontend should support

1. Addition, Modification and Deletion of records from any chosen table
2. There should be an window to accept and run any SQL statement and display the result

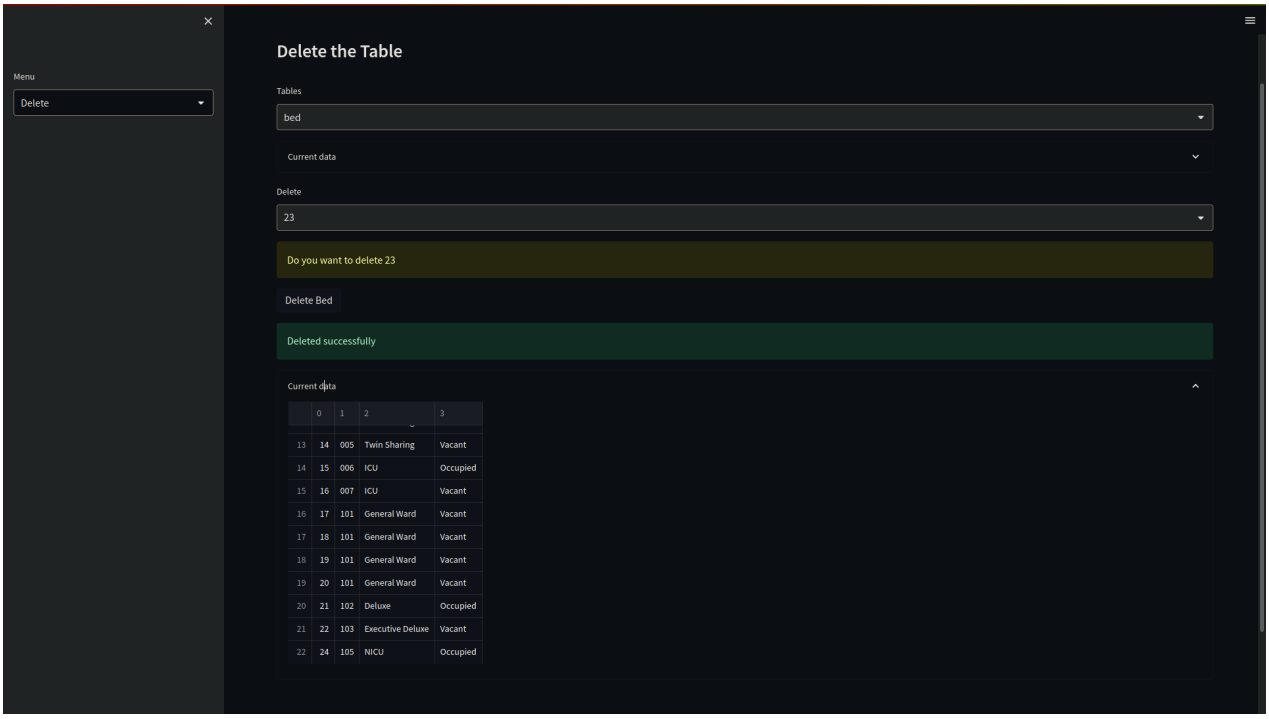
Create



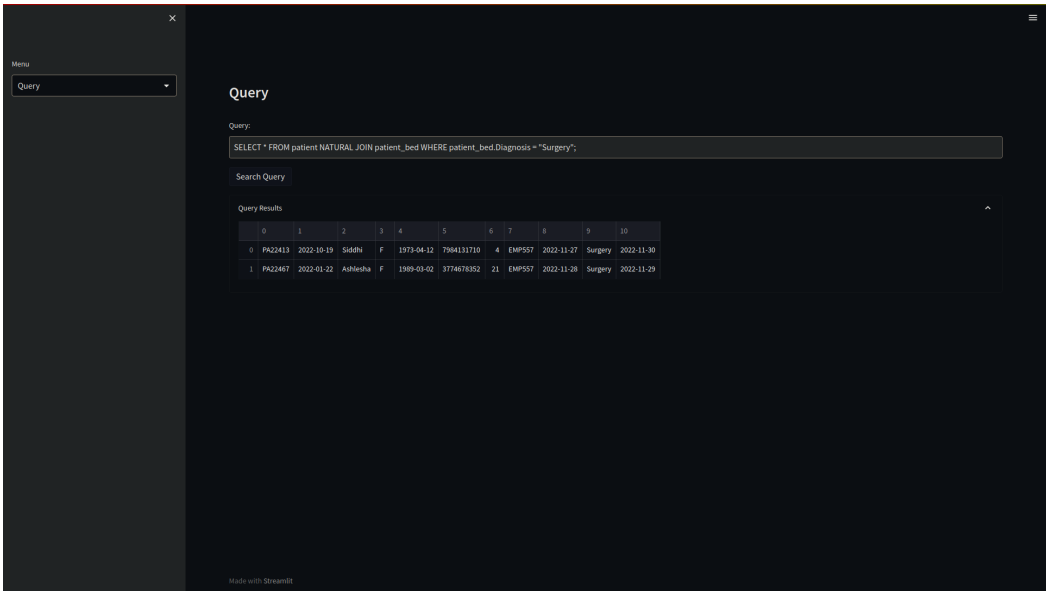
Read



Delete


Menu
Delete
Delete the Table
Tables
bed
Current data
Delete
23
Do you want to delete 23
Delete Bed
Deleted successfully
Current data
0 1 2 3
13 14 005 Twin Sharing Vacant
14 15 006 ICU Occupied
15 16 007 ICU Vacant
16 17 101 General Ward Vacant
17 18 101 General Ward Vacant
18 19 101 General Ward Vacant
19 20 101 General Ward Vacant
20 21 102 Deluxe Occupied
21 22 103 Executive Deluxe Vacant
22 24 105 NICU Occupied

Query


Menu
Query
Query
Query:
SELECT * FROM patient NATURAL JOIN patient_bed WHERE patient_bed.Diagnosis = "Surgery";
Search Query
Query Results
0 1 2 3 4 5 6 7 8 9 10
0 PA22413 2022-10-19 Siddhi F 1973-04-12 7984131710 4 EMP557 2022-11-27 Surgery 2022-11-30
1 PA22467 2022-01-22 Ashlesha F 1989-03-02 3774678352 21 EMP557 2022-11-28 Surgery 2022-11-29
Made with Streamlit