الجامعة السعودية الإلكترونية
SAUDI ELECTRONIC UNIVERSITY
2011-1432

College of Computing and Informatics

Introduction to Database

DS350

Project

Deadline: Thursday 07/12/2023@ 23:59

[Total Mark for this Project is 14]

Stuent Details:

Name: BASSAM NASIB KASAR **ID: S210024849**
Name: Ali Almarhoun **ID: S200066199**
Name: KARRAR HAIDAR **ID: S210029433**

Instructions:

- You must submit two separate copies **(one Word file and one PDF file)** using the Assignment Template on Blackboard via the allocated folder. These files **must not be in compressed format**.
- It is your responsibility to check and make sure that you have uploaded both the correct files.
- Zero mark will be given if you try to bypass the SafeAssign (e.g., misspell words, remove spaces between words, hide characters, use different character sets, convert text into image or languages other than English or any kind of manipulation).
- Email submission will not be accepted.
- You are advised to make your work clear and well-presented. This includes filling your information on the cover page.
- You must use this template, failing which will result in zero mark.
- You MUST show all your work, and text must not be converted into an image, unless specified otherwise by the question.
- Late submission will result in ZERO mark.
- The work should be your own, copying from students or other resources will result in ZERO mark.
- Use **Times New Roman** font for all your answers.

Project Instructions

- You can work on this project as a group (minimum 3 and maximum 4 students). Each group member must submit the project individually with all group member names mentioned in the cover page.
- This project **worth 14 marks** and will be distributed as in the following:

a)	Identify the entity types, attributes, keys.	**(2 marks)**
b)	Identify the relationship and cardinalities.	**(2 marks)**
c)	Draw the ERD.	**(2 marks)**
d)	Schemas before Normalization.	**(1.5 marks)**
e)	Schemas after Normalization.	**(1.5 marks)**
f)	Create the tables.	**(1.5 marks)**
g)	Populate your tables with at least 5 rows.	**(1.5 marks)**
h)	Execute the requested sample queries.	**(2 marks)**

- Each student must submit one report about his/her chosen Project via the Blackboard **(Email submission will not be accepted which will be awarded ZERO marks)**
- Screenshots for answering SQL questions (**f, g, and h**).
- You are advised to make your work clear and well presented; marks may be reduced for poor presentation. This includes filling your information on the cover page.
- You MUST show all your work, and text **must not** be converted into an image, unless specified otherwise by the question.
  A) **Late submission will result in ZERO marks being awarded.**
  B) The work should be your own, **copying from students or other resources will result in ZERO marks.**

Learning Outcome(s):

LO 4

Design a database starting from the conceptual design to the implementation of database schemas.

LO 3

Create Entity-Relationship model, Relational model, and write SQL queries.

Hospital Management System

Year 2030, you are hired as a database developer in local hospital. The top management of the hospital is interested to design a new database to manage their daily routine work. In the requirement identification phase, you have identified five core elements in the hospital

- Patients: The hospital stores crucial patient data, including their personal information (first name, last name, date of birth, gender), contact details (phone number, address), insurance information, and their medical history.
- Doctors: Hospital also maintains records of the medical professionals, including their specialization, contact information, license number, office hours, and email addresses.
- Appointments: For efficient management of the patients visits to the hospitals, the data about each appointment is stores as it connects patients with specific doctors. The appointment date and time, the reason for the visit, and the appointment status (e.g., scheduled, canceled, completed) is also stored.
- Medical Records: A comprehensive record of patients' medical histories is maintained. It includes diagnosis details, treatment plans, prescriptions, test results, and the date of each medical visit. These records are associated with both patients and doctors.
- Medical Staff: Data related to hospital staff members, such as nurses, receptionists, and administrators is also stored. Information like their roles, contact details, shift timings, and salary is recorded for proper staff management.

Hospital Management System

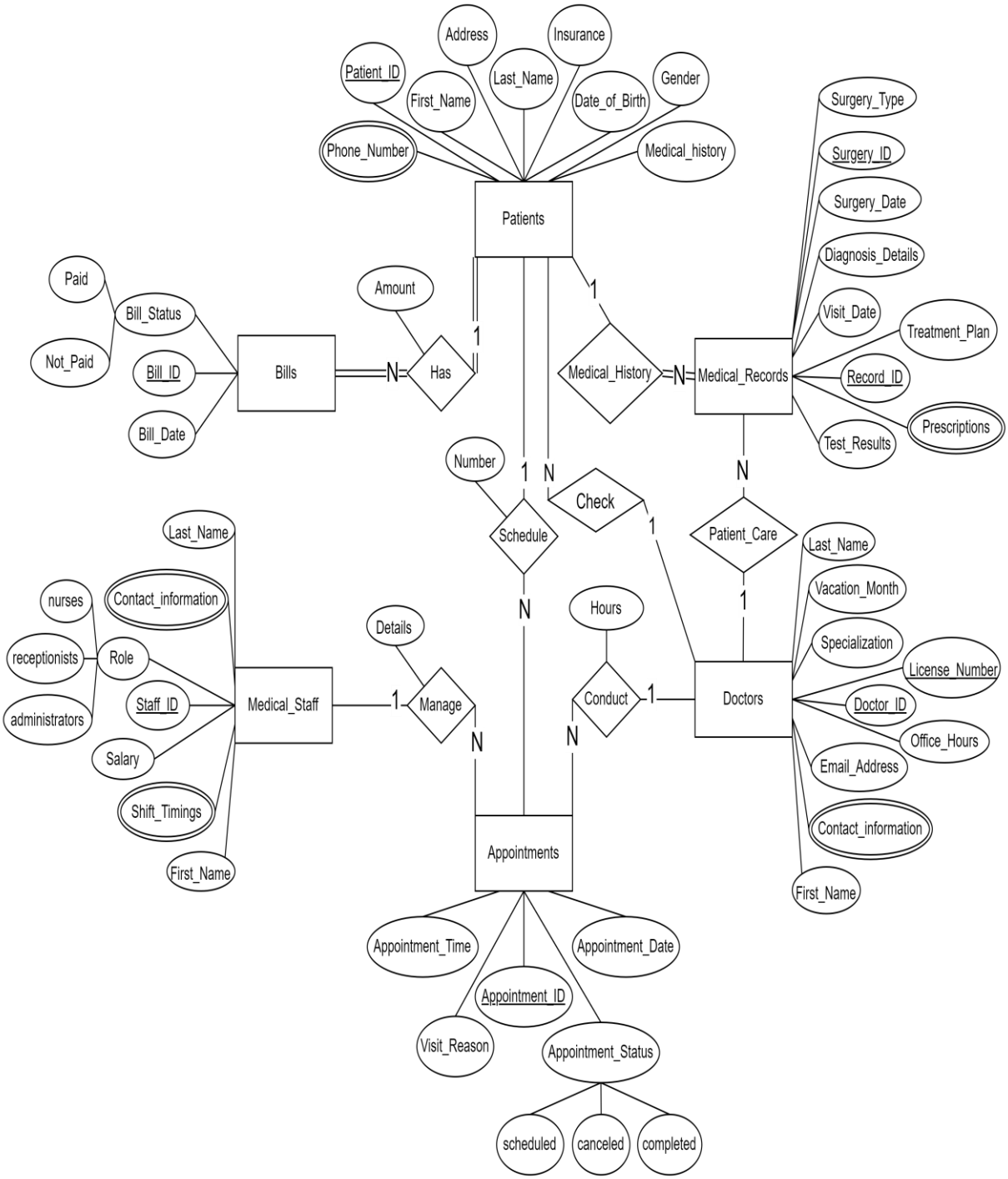
a) Fill the table below with all the Entities based on the given requirements:

Entity Type	Attributes of the Entity	Key
1. Patients	Patient_ID, First_Name, Last_Name, Date_of_Birth, Gender, Phone_Number (Multi-valued), Address, Insurance_information, Medical_history	Patient_ID
2. Doctors	Doctor_ID, First_Name, Last_Name, Specialization, Contact_information (Multi-valued), License_Number, Office_Hours, Email_Address, Vacation_Month	Doctor_ID License_Number
3. Appointments	Appointment_ID, Appointment_Date, Appointment_Time, Visit_Reason, Appointment_Status (scheduled, canceled, completed)	Appointment_ID
4. Medical_Records	Record_ID, Surgery_ID, Surgery_Date, Surgery_Type, Diagnosis_Details, Treatment_Plans, Prescriptions (Multi-valued), Test_Results, Visit_Date	Record_ID Surgery_ID
5. Medical_Staff	Staff_ID, Role (nurses, receptionists, administrators), First_Name, Last_Name, Contact_Information (Multi-valued), Shift_Timings (Multi-valued), Salary	Staff_ID
6. Bills	Bill_ID, Bill_Status (Paid, Not_Paid), Bill_Date	Bill_ID

b) Fill the table below with all the relationship type based on the given requirements.

Relationship Type	Entities connected by the relationship type	Cardinality	Relationship attribute
1. Schedule	Patients and Appointments	One-to-Many	Number
2. Conduct	Doctors - Appointments	One-to-Many	Hours
3. Medical_History	Patients - Medical Records	One-to-Many	
4. Patient_Care	Doctors - Medical Records	One-to-Many	
5. Manage	Medical Staff - Appointments	One-to-Many	Details
6. Has	Patients - Bills	One-to-Many	Amount
7. Check	Doctor – Patients	One-to-Many	

c) Draw the ER diagram



d) Fill the table below with all schemas before normalization.

Schema Name	Schema Attribute	Key
1. Patients	Patient_ID, First_Name, Last_Name, Date_of_Birth, Gender, Address, Insurance_information, Medical_history	Patient_ID
2. Patients_Phone	Patient_ID, Phone_Number	Patient_ID
3. Doctors	Doctor_ID, Patient_ID(FK), First_Name, Last_Name, Specialization, License_Number, Office_Hours, Email_Address, Vacation_Month	Doctor_ID License_Number
4. D_Contact	Doctor_ID, Contact_information	Doctor_ID
5. Appointments	Appointment_ID, Patient_ID(FK), Doctor_ID(FK), Staff_ID(FK), Appointment_Date, Appointment_Time, Visit_Reason, scheduled, canceled, completed	Appointment_ID
6. Medical_Records	Record_ID, Surgery_ID, Patient_ID(FK), Doctor_ID(FK), Surgery_Date, Surgery_Type, Diagnosis_Details, Treatment_Plans, Prescriptions (Multi-valued), Test_Results, Visit_Date	Record_ID Surgery_ID
7. Presc_Records	Record_ID, Prescriptions	Record_ID
8. Medical_Staff	Staff_ID, Nurses, Receptionists, Administrators, Salary	Staff_ID
9. S_Contact	Staff_ID, Contact_Information	Staff_ID
10. Time_Staff	Staff_ID, Shift_Timings	Staff_ID
11. Bills	Bill_ID, Patient_ID(FK), Paid, Not_Paid, Bill_Date	Bill_ID

e) Fill the table below with all schemas after normalization up to the 3NF.

Schema Name	Schema Attribute	Key
1. Patients	Patient_ID, First_Name, Last_Name, Date_of_Birth, Gender, Address, Insurance_information, Medical_history	Patient_ID
2. Patients_Phone	Patient_ID, Phone_Number	Patient_ID
3. Doctors	Doctor_ID, Patient_ID(FK), License_Number(FK), First_Name, Last_Name, Email_Address, Vacation_Month	Doctor_ID
4. License_Number	License_Number, Specialization, Office_Hours	License_Number
5. D_Contact	Doctor_ID, Contact_information	Doctor_ID
6. Appointments	Appointment_ID, Patient_ID(FK), Doctor_ID(FK), Staff_ID(FK), Appointment_Date, Appointment_Time, Visit_Reason, scheduled, canceled, completed	Appointment_ID
7. Medical_Records	Record_ID, Surgery_ID(FK), Patient_ID(FK), Doctor_ID(FK), Diagnosis_Details, Treatment_Plan, Test_Results, Visit_Date	Record_ID
8. Surgeries	Surgery_ID, Patient_ID(FK), Doctor_ID(FK), Surgery_Date, Surgery_Type	Surgery_ID
9. Presc_Records	Record_ID, Prescriptions	Record_ID
10. Medical_Staff	Staff_ID, First_Name, Last_Name, Nurses, Receptionists, Administrators, Salary	Staff_ID
11. S_Contact	Staff_ID, Contact_Information	Staff_ID
12. Time_Staff	Staff_ID, Shift_Timings	Staff_ID
13. Bills	Bill_ID, Patient_ID(FK), Paid, Not_Paid, Bill_Date	Bill_ID

f) Write the necessary SQL statements to create the tables.

```
CREATE TABLE Patients (
    Patient_ID INT PRIMARY KEY,
    First_Name VARCHAR(50),
    Last_Name VARCHAR(50),
    Date_of_Birth DATE,
    Gender VARCHAR(10),
    Address VARCHAR(100),
    Insurance_Information VARCHAR(100),
    Medical_History VARCHAR(200)
);

CREATE TABLE Patients_Phone (
    Patient_ID INT PRIMARY KEY,
    Phone_Number VARCHAR(20)
);

CREATE TABLE Doctors (
    Doctor_ID INT PRIMARY KEY,
    First_Name VARCHAR(50),
    Last_Name VARCHAR(50),
    Patient_ID INT,
    License_Number INT,
    Email_Address VARCHAR(100),
    Vacation_Month VARCHAR(20),
    FOREIGN KEY (Patient_ID) REFERENCES Patients(Patient_ID),
    FOREIGN KEY (License_Number) REFERENCES License_Number(License_Number)
);
```

```
CREATE TABLE License_Number (
    License_Number INT PRIMARY KEY,
    Specialization VARCHAR(50),
    Office_Hours VARCHAR(50)
);

CREATE TABLE D_Contact (
    Doctor_ID INT PRIMARY KEY,
    Contact_Information VARCHAR(100)
);

CREATE TABLE Appointments (
    Appointment_ID INT PRIMARY KEY,
    Patient_ID INT,
    Doctor_ID INT,
    Staff_ID INT,
    Appointment_Date DATE,
    Appointment_Time TIME,
    Visit_Reason VARCHAR(200),
    Scheduled BOOLEAN,
    Canceled BOOLEAN,
    Completed BOOLEAN,
    FOREIGN KEY (Patient_ID) REFERENCES Patients(Patient_ID),
    FOREIGN KEY (Doctor_ID) REFERENCES Doctors(Doctor_ID),
    FOREIGN KEY (Staff_ID) REFERENCES Medical_Staff(Staff_ID)
);
```

```
CREATE TABLE Medical_Records (
    Record_ID INT PRIMARY KEY,
    Patient_ID INT,
    Doctor_ID INT,
    Surgery_ID INT,
    Diagnosis_Details VARCHAR(200),
    Treatment_Plans VARCHAR(200),
    Test_Results VARCHAR(200),
    Visit_Date DATE,
    FOREIGN KEY (Patient_ID) REFERENCES Patients(Patient_ID),
    FOREIGN KEY (Doctor_ID) REFERENCES Doctors(Doctor_ID)
    FOREIGN KEY (Surgery_ID) REFERENCES Surgeries(Surgery_ID)
);

CREATE TABLE Surgeries (
    Surgery_ID INT PRIMARY KEY,
    Doctor_ID INT,
    Patient_ID INT,
    Surgery_Date DATE,
    Surgery_Type VARCHAR(100),
    FOREIGN KEY (Doctor_ID) REFERENCES Doctors(Doctor_ID),
    FOREIGN KEY (Patient_ID) REFERENCES Patients(Patient_ID)
);

CREATE TABLE Presc_Records (
    Record_ID INT PRIMARY KEY,
    Prescriptions VARCHAR(200),
    FOREIGN KEY (Record_ID) REFERENCES Medical_Records(Record_ID)
);
```

```
CREATE TABLE Medical_Staff (
    Staff_ID INT PRIMARY KEY,
    First_Name VARCHAR(50),
    Last_Name VARCHAR(50),
    Nurses BOOLEAN,
    Receptionists BOOLEAN,
    Administrators BOOLEAN,
    Salary DECIMAL(10, 2)
);

CREATE TABLE S_Contact (
    Staff_ID INT PRIMARY KEY,
    Contact_Information VARCHAR(100),
    FOREIGN KEY (Staff_ID) REFERENCES Medical_Staff(Staff_ID)
);

CREATE TABLE Time_Staff (
    Staff_ID INT PRIMARY KEY,
    Shift_Timings VARCHAR(100),
    FOREIGN KEY (Staff_ID) REFERENCES Medical_Staff(Staff_ID)
);

CREATE TABLE Bills (
    Bill_ID INT PRIMARY KEY,
    Patient_ID INT,
    Paid DECIMAL(10, 2),
    Not_Paid DECIMAL(10, 2),
    Bill_Date DATE,
    FOREIGN KEY (Patient_ID) REFERENCES Patients(Patient_ID)
);
```

g) Insert at least five rows into each table.

```
-- insert some values

INSERT INTO Patients (Patient_ID, First_Name, Last_Name, Date_of_Birth, Gender,
Address, Insurance_Information, Medical_History) VALUES
(1, 'Ali', 'khodor', '1990-05-15', 'Male', '123 Main St', 'ABC Insurance', 'Heart condition'),
(2, 'Lyla', 'Al-gamdi', '1985-08-20', 'Female', '456 Elm St', 'XYZ Insurance', 'Allergies'),
(3, 'Fatima', 'Al-dosary', '1978-12-10', 'Female', '789 Oak St', 'PQR Insurance', 'Diabetes'),
(4, 'Ahmed', 'Al-Abdullah', '1982-07-25', 'Male', '101 Pine St', 'LMN Insurance', 'Asthma'),
(5, 'Mariam', 'Al-dosary', '1995-03-08', 'Female', '222 Cedar St', 'STU Insurance', 'Migraines');

-- Sample data for Patients_Phone table

INSERT INTO Patients_Phone (Patient_ID, Phone_Number) VALUES
(1, '123-456-7890'),
(2, '987-654-3210'),
(3, '111-222-3333'),
(4, '444-555-6666'),
(5, '777-888-9999');

-- Sample data for Doctors table

INSERT INTO Doctors (Doctor_ID, First_Name, Last_Name, Patient_ID,
License_Number, Email_Address, Vacation_Month) VALUES
(101, 'Ibrahim', 'Al-own', 1, 1001, 'Ibrahim.Al-own@example.com', 'January'),
(102, 'Natheer', 'Maki', 2, 1002, 'Natheer@example.com', 'March'),
(103, 'Hassan', 'Al-dahneem', 3, 1003, 'hassan@example.com', 'July'),
(104, 'Taqi', 'Al-ajami', 4, 1004, 'taqi@example.com', 'August'),
(105, 'Abdullah', 'Al-owa', 5, 1005, 'abdullah@example.com', 'October');

-- Sample data for License_Number table

INSERT INTO License_Number (License_Number, Specialization, Office_Hours)
VALUES
(1001, 'Cardiology', '9 AM - 5 PM'),
(1002, 'Pediatrics', '10 AM - 6 PM'),
(1003, 'Endocrinology', '8 AM - 4 PM'),
(1004, 'Pulmonology', '9:30 AM - 5:30 PM'),
(1005, 'Neurology', '8:30 AM - 4:30 PM');
```

```
-- Sample data for D_Contact table

INSERT INTO D_Contact (Doctor_ID, Contact_Information) VALUES
(101, 888-999-1111),
(102, 999-111-2222),
(103, 101-202-3003),
(104, 202-303-4004),
(105, 303-404-5005);

-- Sample data for Appointments table

INSERT INTO Appointments (Appointment_ID, Patient_ID, Doctor_ID, Staff_ID,
Appointment_Date, Appointment_Time, Visit_Reason, Scheduled, Canceled,
Completed) VALUES
(201, 1, 101, 301, '2023-09-10', '10:00:00', 'Regular checkup', true, false, false),
(202, 2, 102, 302, '2023-09-15', '11:30:00', 'Sore throat', true, false, false),
(203, 3, 103, 303, '2023-09-20', '09:00:00', 'Insulin dosage', true, false, false),
(204, 4, 104, 304, '2023-09-25', '13:45:00', 'Breathing issues', true, false, false),
(205, 5, 105, 305, '2023-09-30', '19:45:00', 'Headaches', true, false, false);

-- Sample data for Medical_Records table

INSERT INTO Medical_Records (Record_ID, Patient_ID, Doctor_ID, Surgery_ID,
Diagnosis_Details, Treatment_Plans, Test_Results, Visit_Date) VALUES
(301, 1, 101, 501, 'Routine checkup', 'Prescribed medication', 'Normal', '2023-09-10'),
(302, 2, 102, 502, 'Tonsillitis', 'Prescribed antibiotics', 'Negative', '2023-09-15'),
(303, 3, 103, 503, 'Diabetes management', 'Insulin regimen', 'Stable', '2023-09-20'),
(304, 4, 104, 504, 'Asthma exacerbation', 'Inhaled corticosteroids', 'Improved', '2023-09-25'),
(305, 5, 105, 505, 'Migraine headaches', 'Prescribed pain relief', 'Moderate', '2023-09-30');

-- Sample data for Presc_Records table

INSERT INTO Presc_Records (Record_ID, Prescriptions) VALUES
(301, 'Medication A'),
(302, 'Antibiotic B'),
(303, 'Insulin C'),
(304, 'Inhaler D'),
(305, 'Pain Reliever E');
```

```
-- Sample data for Medical_Staff table

INSERT INTO Medical_Staff (Staff_ID, First_Name, Last_Name, Nurses,
Receptionists, Administrators, Salary) VALUES
(301, 'Mohammed', 'al-Ahmed ', true, false, false, 50000),
(302, 'Faisal', 'al-Bassam', false, true, false, 45000),
(303, 'Ahmed', 'al-Mohammed', false, false, true, 60000),
(304, 'Saleh', 'al-Salah', true, false, false, 52000),
(305, 'Azam', 'al-Faisal', false, true, false, 47000);

-- Sample data for S_Contact table

INSERT INTO S_Contact (Staff_ID, Contact_Information) VALUES
(301, '123-456-7890'),
(302, '222-333-4444'),
(303, '555-666-7777'),
(304, '888-999-0000'),
(305, '111-222-3333');

-- Sample data for Time_Staff table

INSERT INTO Time_Staff (Staff_ID, Shift_Timings) VALUES
(301, 'Day Shift'),
(302, 'Evening Shift'),
(303, 'Night Shift'),
(304, 'Day Shift'),
(305, 'Evening Shift');

-- Sample data for Bills table

INSERT INTO Bills (Bill_ID, Patient_ID, Paid, Not_Paid, Bill_Date) VALUES
(401, 1, 150.00, 50.00, '2023-09-12'),
(402, 2, 200.00, 0.00, '2023-09-18'),
(403, 3, 300.00, 0.00, '2023-09-22'),
(404, 4, 250.00, 100.00, '2023-09-28'),
(405, 5, 180.00, 20.00, '2023-10-03');
```

```
-- Sample data for Surgeries table

INSERT INTO Surgeries (Surgery_ID, Doctor_ID, Patient_ID, Surgery_Date,
Surgery_Type) VALUES
(501, 101, 1, '2023-08-20', 'Heart Surgery'),
(502, 102, 2, '2023-09-05', 'Tonsillectomy'),
(503, 103, 3, '2023-09-15', 'Insulin Pump Insertion'),
(504, 104, 4, '2023-09-25', 'Lung Biopsy'),
(505, 105, 5, '2023-09-29', 'Brain MRI'),
(506, 105, 5, '2023-10-02', 'Brain MRI'),
(507, 105, 5, '2023-10-04', 'Brain MRI'),
(508, 105, 5, '2023-10-05', 'Brain MRI'),
(509, 105, 5, '2023-10-06', 'Brain MRI'),
(510, 105, 5, '2023-10-02', 'Brain MRI'),
(511, 105, 5, '2023-10-07', 'Brain MRI'),
(512, 105, 5, '2023-10-08', 'Brain MRI'),
(513, 105, 5, '2023-10-08', 'Brain MRI'),
(514, 105, 5, '2023-10-08', 'Brain MRI'),
(515, 105, 5, '2023-10-09', 'Brain MRI'),
(516, 105, 5, '2023-10-12', 'Brain MRI');
```

Note: We add Bills entity. Surgery_ID, Surgery_Date, Surgery_Type and Vacation_Month attributes for answering Section h: Q1, Q2, Q3 and Q4

h) Write SQL queries to find the following:

1) List of Doctors who have performed more than 10 surgeries.

```
SELECT Doctors.Doctor_ID, First_Name, Last_Name
FROM Doctors
JOIN Surgeries ON Doctors.Doctor_ID = Surgeries.Doctor_ID
GROUP BY Doctors.Doctor_ID
HAVING COUNT(Surgeries.Surgery_ID) > 10;
```

Output

```
105|Abdullah|Al-owa

[Execution complete with exit code 0]
```

2) List of Patients admitted for surgery between 1/9/2023 and 31/9/2023.

```
SELECT DISTINCT Patients.Patient_ID, First_Name, Last_Name
FROM Patients
JOIN Surgeries ON Patients.Patient_ID = Surgeries.Patient_ID
WHERE Surgery_Date BETWEEN '2023-09-01' AND '2023-09-30';
```

Output

```
2|Lyla|Al-gamdi
3|Fatima|Al-dosary
4|Ahmed|Al-Abdullah
5|Mariam|Al-dosary

[Execution complete with exit code 0]
```

3) List of all the Patients who have not paid their bills till today.

```
SELECT Patients.Patient_ID, First_Name, Last_Name

FROM Patients

LEFT JOIN Bills ON Patients.Patient_ID = Bills.Patient_ID

WHERE Not_Paid > 0 OR Not_Paid IS NULL;
```

Output

```
1|Ali|khodor
4|Ahmed|Al-Abdullah
5|Mariam|Al-dosary

[Execution complete with exit code 0]
```

4) List of Doctors on leave in a particular month.

```
SELECT Doctor_ID, First_Name, Last_Name, Vacation_Month

FROM Doctors

WHERE Vacation_Month = 'August';
```

Output

```
104|Taqi|Al-ajami|August

[Execution complete with exit code 0]
```

5) List of Appointments with each Doctor with Patient name for a particular day.

```
SELECT Appointments.Appointment_ID,
Doctors.First_Name AS Doctor_First_Name,
Doctors.Last_Name AS Doctor_Last_Name,

Patients.First_Name AS Patient_First_Name,
Patients.Last_Name AS Patient_Last_Name,

Appointments.Appointment_Date, Appointments.Appointment_Time

FROM Appointments

JOIN Doctors ON Appointments.Doctor_ID = Doctors.Doctor_ID

JOIN Patients ON Appointments.Patient_ID = Patients.Patient_ID

WHERE Appointments.Appointment_Date = '2023-09-15';
```

Output

```
202|Natheer|Maki|Lyla|Al-gamdi|2023-09-15|11:30:00

[Execution complete with exit code 0]
```

6) Analyze Using Business Intelligence Power BI: