

# Database Project



**Presented To:**

***Prof. Hoda Korashy Mohamed***

***Course Code: CSE333s***

***Course Name: Database Systems***

**Ain Shams University  
Faculty of Engineering**



## Database Project

“Hospital”

Name	ID
Mazen Ehab Mohamed Maher Osman Hekal	1901120
Ahmed Mahmoud Mohamed Ibrahim	1901143
Mohamed Mostafa Shaban Mohamed	1901650
Andrew Samir Kamel Gayed	1900242



Contents:

Domain.....	04
Introduction and Description.....	04
Important Data and Reports.....	05
• Data.....	05
• Reports.....	10
Assumptions.....	11
EER Diagram.....	12
Database Schema.....	13
Sample of SQL.....	14
• SQL Create Tables.....	14
• SQL Insertions.....	18
• SQL Updates and Deletes.....	25
• SQL Report Retrievals.....	27
Implementation.....	34
• Part 1: using ERD Tool.....	34
○ EER Diagram.....	35
○ Relational Schema.....	36
○ SQL.....	37
• Part 2: using SQL Tool.....	41
○ Example for Create.....	41
○ Example for Insert.....	42
○ Example for Update.....	42
○ Example for Delete.....	42
○ Example for Select.....	43



## Domain

### “Hospital”

#### 1. Introduction and Description

This database represents a hospital management system. It includes tables for storing information about patients, employees, rooms, operations, and other entities. It allows for efficient management of hospital resources, including medical care rooms, operation rooms, and clinics, as well as the staff and patients who occupy them. Overall, this database provides a comprehensive solution for managing hospital resources and ensuring efficient patient care.

The following are some of the features of the database:

- It can store information about patients, including their name, address, date of birth, gender, contact information, medical history, and insurance information.
- It can store information about employees, including their name, job title, department, salary, and contact information.
- It can store information about rooms, including their number, floor, type, and capacity.
- It can store information about operations, including the name of the operation, the date of the operation, the room where the operation was performed, the doctors and nurses who performed the operation.
- It can store information about other entities, such as doctors, nurses, clinics, departments, and medical records.

The database is designed to be easy to use and maintain. It uses a relational database design, which makes it easy to query and update data. The database is also scalable, so it can be easily expanded to accommodate the growth of the hospital.



## 2. Important Data and Reports

### Data:

#### 1. Person

Name	Data Type
SSN	INT
FNAME	STRING
LNAME	STRING
BirthDate	DATE
Gender	STRING
Address	STRING

#### 2. Person\_PhoneNo

Name	Data Type
SSN	INT
PhoneNo	INT

#### 3. Patient

Name	Data Type
SSN	INT
Diagnosis	STRING
PatientID	INT

#### 4. MedicalRecord

Name	Data Type
MedicalRecordID	INT
ExaminationDate	DATE
Description	STRING
PatientSSN	INT



5. Employee

Name	Data Type
SSN	INT
EmployeeID	INT
Salary	INT
WorkingHours	INT

6. Room

Name	Data Type
RoomID	INT
FloorNo	INT

7. MedicalCareRoom

Name	Data Type
RoomID	INT
BedsCount	INT

8. OperationRoom

Name	Data Type
RoomID	INT
MaxDoctorsNo	INT

9. OperationRoom\_Equipment

Name	Data Type
OperationRoomID	INT
Equipment	STRING



### 10. Receptionist

Name	Data Type
SSN	INT
DeskNo	INT

### 11. Admin

Name	Data Type
SSN	INT
AdminID	INT

### 12. OutPatient

Name	Data Type
SSN	INT
ReservationNo	INT

### 13. InPatient

Name	Data Type
SSN	INT
AdmissionDate	DATE
DischargeDate	DATE
OperationID	INT
RoomID	INT



**14. Clinic**

Name	Data Type
ClinicID	INT
ClinicType	STRING
FloorNo	INT

**15. VISITS**

Name	Data Type
OutPatientSSN	INT
ClinicID	INT
VisitDate	DATE

**16. Operation**

Name	Data Type
OperationID	INT
OperationName	STRING
OperationDate	DATE
OperationRoomID	INT

**17. Department**

Name	Data Type
DepartmentID	INT
DepartmentName	STRING
MGRSSN	INT





**18. Doctor**

Name	Data Type
SSN	INT
Speciality	STRING
DepartmentID	INT
ClinicID	INT

**19. Nurse**

Name	Data Type
SSN	INT
NursingService	STRING
DepartmentID	INT

**20. PERFORMS**

Name	Data Type
DoctorSSN	INT
OperationID	INT



**Reports:**

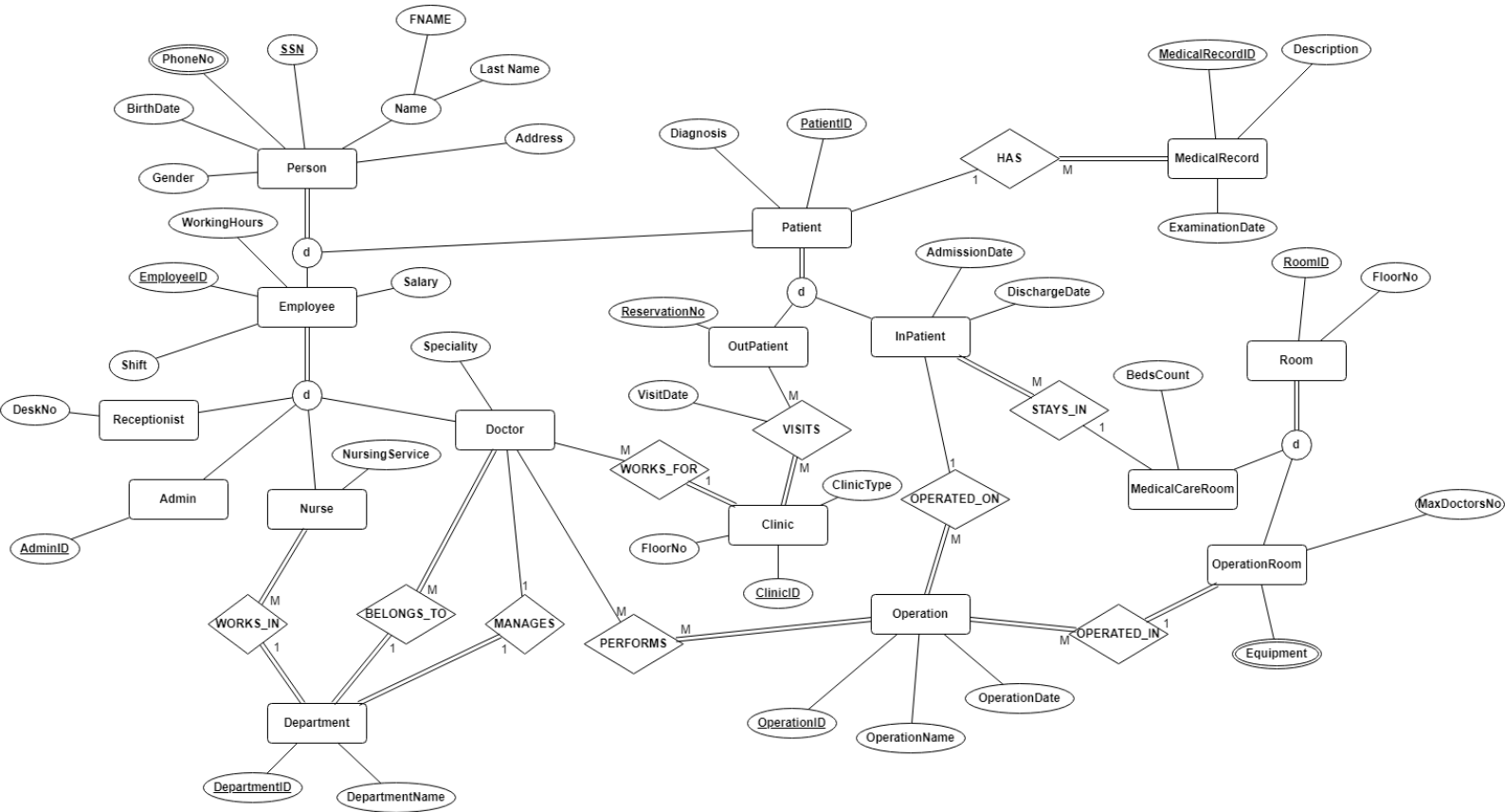
- Get all doctors who manage a department
- Get count of doctors and nurses in each department
- Get number of all inpatients who stayed in each medical care room
- Get average and total salary for each department
- Get doctors who work in a clinic
- Get all doctors who have performed at least one operation
- Get all persons who work as receptionist
- Get Phone No. for every nurse
- Get medical record for every inpatient
- Get count of inpatients in hospital
- Get count of receptionists in hospital
- Get No of Doctors in each Clinic
- Get the name and description of medical records associated with a patient who has a reservation at a particular Clinic
- Get the names and admission dates of inpatients who have not yet been discharged



### 3. Assumptions

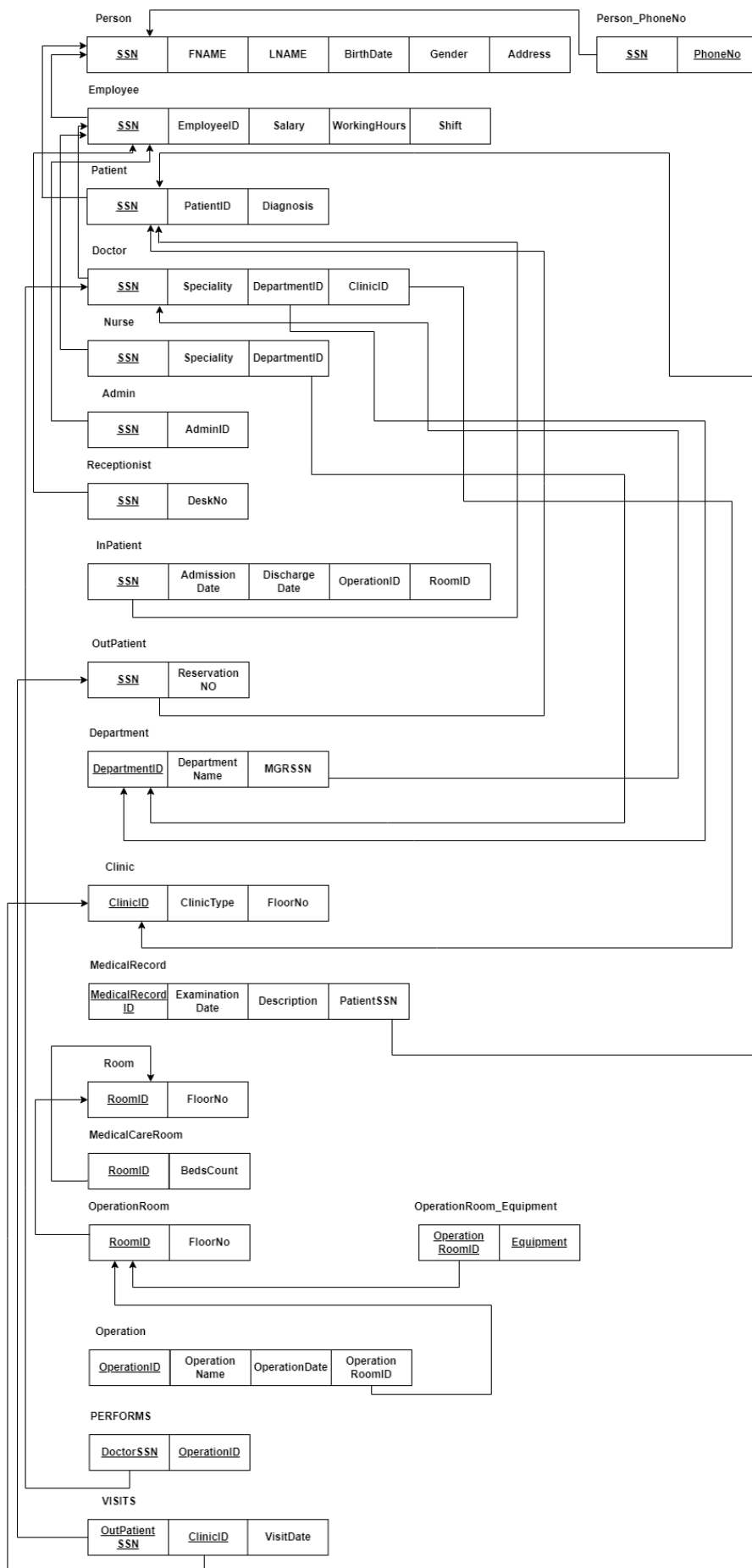
1. A person must be either an employee or a patient.
2. An employee is only a receptionist, an admin, a nurse, or a doctor.
3. A patient must be either an outpatient or an inpatient.
4. In the hospital, a room must be either an operation room or a medical care room.
5. A nurse must work in one department only, and each department must have one or many nurses.
6. A doctor must belong to one department only, and each department must have one or many doctors.
7. A doctor may manage only one department, and each department must be managed by one doctor only (head of department).
8. A doctor may perform one or many operations, and an operation must be performed by one or many doctors.
9. A patient (inpatient or outpatient) may have one or many medical records, and each medical record is for one patient only.
10. An inpatient must stay in only one medical care room, and each medical care room may have one or many inpatients.
11. An operation must be performed on only one inpatient, and each inpatient may undergo one or many operations.
12. A doctor may work for only one clinic, and each clinic must have one or many doctors.
13. An outpatient may visit one or many clinics, and each clinic may be visited by one or many outpatients.
14. An operation must be performed in only one operation room, and each operation room may accommodate one or many operations.

## 4. EER Diagram





## 5. Database Schema





## 6. Sample Of SQL

- **SQL Create Tables**

```
CREATE TABLE Person
```

```
(  
  SSN INT NOT NULL,  
  FNAME VARCHAR(50) NOT NULL,  
  LNAME VARCHAR(50) NOT NULL,  
  BirthDate DATE NOT NULL,  
  Gender CHAR(1) NOT NULL,  
  Address VARCHAR(100) NOT NULL,  
  PRIMARY KEY (SSN)  
);
```

```
CREATE TABLE Patient
```

```
(  
  SSN INT NOT NULL,  
  Diagnosis VARCHAR(50) NOT NULL,  
  PatientID INT NOT NULL,  
  PRIMARY KEY (SSN),  
  FOREIGN KEY (SSN) REFERENCES Person(SSN)  
  ON DELETE CASCADE  
  ON UPDATE CASCADE,  
  UNIQUE (PatientID)  
);
```

```
CREATE TABLE Employee
```

```
(  
  SSN INT NOT NULL,  
  EmployeeID INT NOT NULL,  
  Salary INT NOT NULL,  
  WorkingHours INT NOT NULL,  
  Shift VARCHAR(50) NOT NULL,  
  PRIMARY KEY (SSN),  
  FOREIGN KEY (SSN) REFERENCES Person(SSN)  
  ON DELETE CASCADE  
  ON UPDATE CASCADE,  
  UNIQUE (EmployeeID)  
);
```

```
CREATE TABLE Room
```

```
(  
  RoomID INT NOT NULL,  
  FloorNo INT NOT NULL,  
  PRIMARY KEY (RoomID)  
);
```



```
CREATE TABLE Receptionist
(
    SSN INT NOT NULL,
    DeskNo INT NOT NULL,
    PRIMARY KEY (SSN),
    FOREIGN KEY (SSN) REFERENCES Employee(SSN)
    ON DELETE CASCADE
    ON UPDATE CASCADE
);

CREATE TABLE MedicalCareRoom
(
    RoomID INT NOT NULL,
    BedsCount INT NOT NULL,
    PRIMARY KEY (RoomID),
    FOREIGN KEY (RoomID) REFERENCES Room(RoomID)
    ON DELETE CASCADE
    ON UPDATE CASCADE
);

CREATE TABLE OperationRoom
(
    RoomID INT NOT NULL,
    MaxDoctorsNo INT NOT NULL,
    PRIMARY KEY (RoomID),
    FOREIGN KEY (RoomID) REFERENCES Room(RoomID)
    ON DELETE CASCADE
    ON UPDATE CASCADE
);

CREATE TABLE MedicalRecord
(
    MedicalRecordID INT NOT NULL,
    ExaminationDate DATE NOT NULL,
    Description VARCHAR(150) NOT NULL,
    PatientSSN INT NOT NULL,
    PRIMARY KEY (MedicalRecordID),
    FOREIGN KEY (PatientSSN) REFERENCES Patient(SSN)
    ON DELETE CASCADE
    ON UPDATE CASCADE
);

CREATE TABLE Admin
(
    SSN INT NOT NULL,
    AdminID INT NOT NULL,
    PRIMARY KEY (SSN),
    FOREIGN KEY (SSN) REFERENCES Employee(SSN)
    ON DELETE CASCADE
    ON UPDATE CASCADE,
    UNIQUE (AdminID)
);
```



```
CREATE TABLE OutPatient
(
    SSN INT NOT NULL,
    ReservationNo INT NOT NULL,
    PRIMARY KEY (SSN),
    FOREIGN KEY (SSN) REFERENCES Patient(SSN)
    ON DELETE CASCADE
    ON UPDATE CASCADE,
    UNIQUE (ReservationNo)
);

CREATE TABLE Clinic
(
    ClinicID INT NOT NULL,
    ClinicType VARCHAR(50) NOT NULL,
    FloorNo INT NOT NULL,
    PRIMARY KEY (ClinicID)
);

CREATE TABLE VISITS
(
    OutPatientSSN INT NOT NULL,
    ClinicID INT NOT NULL,
    VisitDate DATE NOT NULL,
    PRIMARY KEY (OutPatientSSN, ClinicID),
    FOREIGN KEY (OutPatientSSN) REFERENCES OutPatient(SSN)
    ON DELETE RESTRICT
    ON UPDATE CASCADE,
    FOREIGN KEY (ClinicID) REFERENCES Clinic(ClinicID)
    ON DELETE CASCADE
    ON UPDATE CASCADE
);

CREATE TABLE Person_PhoneNo
(
    SSN INT NOT NULL,
    PhoneNo INT NOT NULL,
    PRIMARY KEY (PhoneNo, SSN),
    FOREIGN KEY (SSN) REFERENCES Person(SSN)
    ON DELETE CASCADE
    ON UPDATE CASCADE
);

CREATE TABLE OperationRoom_Equipment
(
    OperationRoomID INT NOT NULL,
    Equipment VARCHAR(50) NOT NULL,
    PRIMARY KEY (Equipment, OperationRoomID),
    FOREIGN KEY (OperationRoomID) REFERENCES OperationRoom(RoomID)
    ON DELETE CASCADE
    ON UPDATE CASCADE
);
```





```
CREATE TABLE Operation
(
  OperationID INT NOT NULL,
  OperationName VARCHAR(50) NOT NULL,
  OperationDate DATE NOT NULL,
  OperationRoomID INT NOT NULL,
  PRIMARY KEY (OperationID),
  FOREIGN KEY (OperationRoomID) REFERENCES OperationRoom(RoomID)
  ON DELETE RESTRICT
  ON UPDATE CASCADE
);
```

```
CREATE TABLE InPatient
(
  SSN INT NOT NULL,
  AdmissionDate DATE NOT NULL,
  DischargeDate DATE,
  OperationID INT,
  RoomID INT NOT NULL,
  PRIMARY KEY (SSN),
  FOREIGN KEY (SSN) REFERENCES Patient(SSN)
  ON DELETE CASCADE
  ON UPDATE CASCADE,
  FOREIGN KEY (OperationID) REFERENCES Operation(OperationID)
  ON DELETE RESTRICT
  ON UPDATE CASCADE,
  FOREIGN KEY (RoomID) REFERENCES MedicalCareRoom(RoomID)
  ON DELETE RESTRICT
  ON UPDATE CASCADE
);
```

```
CREATE TABLE Department
(
  DepartmentID INT NOT NULL,
  DepartmentName VARCHAR(50) NOT NULL,
  MGRSSN INT,
  PRIMARY KEY (DepartmentID)
);
```

```
CREATE TABLE Doctor
(
  SSN INT NOT NULL,
  Speciality VARCHAR(50) NOT NULL,
  DepartmentID INT NOT NULL,
  ClinicID INT,
  PRIMARY KEY (SSN),
  FOREIGN KEY (SSN) REFERENCES Employee(SSN)
  ON DELETE CASCADE
  ON UPDATE CASCADE,
  FOREIGN KEY (DepartmentID) REFERENCES Department(DepartmentID)
  ON DELETE RESTRICT
  ON UPDATE CASCADE,
  FOREIGN KEY (ClinicID) REFERENCES Clinic(ClinicID)
  ON DELETE SET NULL
  ON UPDATE CASCADE
);
```



```
CREATE TABLE Nurse
(
    SSN INT NOT NULL,
    NursingService VARCHAR(100) NOT NULL,
    DepartmentID INT NOT NULL,
    PRIMARY KEY (SSN),
    FOREIGN KEY (SSN) REFERENCES Employee(SSN)
    ON DELETE CASCADE
    ON UPDATE CASCADE,
    FOREIGN KEY (DepartmentID) REFERENCES Department(DepartmentID)
    ON DELETE RESTRICT
    ON UPDATE CASCADE
);
```

```
CREATE TABLE PERFORMS
(
    DoctorSSN INT NOT NULL,
    OperationID INT NOT NULL,
    PRIMARY KEY (DoctorSSN, OperationID),
    FOREIGN KEY (DoctorSSN) REFERENCES Doctor(SSN)
    ON DELETE RESTRICT
    ON UPDATE CASCADE,
    FOREIGN KEY (OperationID) REFERENCES Operation(OperationID)
    ON DELETE RESTRICT
    ON UPDATE CASCADE
);
```

- **SQL Insertions**

```
INSERT INTO Person (SSN, FNAME, LNAME, BirthDate, Gender, Address) VALUES
(301123456, 'John', 'Doe', '1990-01-01', 'M', '123 Main St'),
(301234567, 'Jane', 'Doe', '1995-02-02', 'F', '456 Elm St'),
(301345678, 'Bob', 'Smith', '1985-03-03', 'M', '789 Oak St'),
(301456789, 'Sarah', 'Johnson', '1992-04-04', 'F', '321 Pine St'),
(301567890, 'Mike', 'Williams', '1988-05-05', 'M', '654 Cedar St'),
(301678901, 'Emily', 'Brown', '1998-06-06', 'F', '987 Maple St'),
(301789012, 'David', 'Lee', '1980-07-07', 'M', '246 Birch St'),
(301890123, 'Karen', 'Davis', '1983-08-08', 'F', '369 Walnut St'),
(301901234, 'Tom', 'Miller', '1999-09-09', 'M', '582 Oak St'),
(301112233, 'Linda', 'Taylor', '1991-10-10', 'F', '753 Elm St'),
(301223344, 'Chris', 'Anderson', '1986-11-11', 'M', '951 Pine St'),
(301334455, 'Jenny', 'Wilson', '1997-12-12', 'F', '864 Cedar St'),
(301445566, 'Kevin', 'Davis', '1982-01-13', 'M', '753 Maple St'),
(301556677, 'Rachel', 'Johnson', '1994-02-14', 'F', '369 Birch St'),
(301667788, 'Brian', 'Lee', '1984-03-15', 'M', '582 Walnut St'),
(301778899, 'Laura', 'Brown', '1996-04-16', 'F', '951 Oak St'),
(301889900, 'Mark', 'Miller', '1981-05-17', 'M', '753 Elm St'),
(301990011, 'Samantha', 'Taylor', '1993-06-18', 'F', '864 Pine St'),
(301102030, 'Alex', 'Anderson', '2000-07-19', 'M', '753 Cedar St'),
(301203040, 'Jessica', 'Wilson', '1987-08-20', 'F', '369 Maple St'),
(301304050, 'Matt', 'Davis', '1990-09-21', 'M', '582 Birch St'),
(301405060, 'Olivia', 'Johnson', '1992-10-22', 'F', '951 Walnut St'),
(301506070, 'Chris', 'Lee', '1989-11-23', 'M', '753 Oak St'),
(301607080, 'Nicole', 'Brown', '1991-12-24', 'F', '864 Elm St'),
(301708090, 'Adam', 'Miller', '1979-01-25', 'M', '753 Pine St'),
(301809010, 'Emily', 'Taylor', '1998-02-26', 'F', '369 Cedar St'),
(301910203, 'Mike', 'Anderson', '1984-03-27', 'M', '582 Maple St'),
```



(301112244, 'Julia', 'Wilson', '1995-04-28', 'F', '951 Birch St'),  
(301223346, 'Tommy', 'Davis', '1980-05-29', 'M', '753 Walnut St'),  
(301334458, 'Hannah', 'Johnson', '1997-06-30', 'F', '864 Oak St'),  
(301334789, 'Anakin', 'Shephard', '2001-05-25', 'M', '588 Maple St');

-- Insert two unique phone numbers for John Doe  
INSERT INTO Person\_PhoneNo (SSN, PhoneNo)  
VALUES (301123456, 1234567890),  
(301123456, 1234567891);

-- Insert two unique phone numbers for Jane Doe  
INSERT INTO Person\_PhoneNo (SSN, PhoneNo)  
VALUES (301234567, 1234567892),  
(301234567, 1234567893);

-- Insert two unique phone numbers for Bob Smith  
INSERT INTO Person\_PhoneNo (SSN, PhoneNo)  
VALUES (301345678, 1234567894),  
(301345678, 1234567895);

-- Insert two unique phone numbers for Sarah Johnson  
INSERT INTO Person\_PhoneNo (SSN, PhoneNo)  
VALUES (301456789, 1234567896),  
(301456789, 1234567897);

-- Insert two unique phone numbers for Mike Williams  
INSERT INTO Person\_PhoneNo (SSN, PhoneNo)  
VALUES (301567890, 1234567898),  
(301567890, 1234567899);

-- Insert two unique phone numbers for Emily Brown  
INSERT INTO Person\_PhoneNo (SSN, PhoneNo)  
VALUES (301678901, 1234567900),  
(301678901, 1234567901);

-- Insert two unique phone numbers for David Lee  
INSERT INTO Person\_PhoneNo (SSN, PhoneNo)  
VALUES (301789012, 1234567902),  
(301789012, 1234567903);

-- Insert two unique phone numbers for Karen Davis  
INSERT INTO Person\_PhoneNo (SSN, PhoneNo)  
VALUES (301890123, 1234567904),  
(301890123, 1234567905);

-- Insert two unique phone numbers for Tom Miller  
INSERT INTO Person\_PhoneNo (SSN, PhoneNo)  
VALUES (301901234, 1234567906),  
(301901234, 1234567907);

-- Insert two unique phone numbers for Linda Taylor  
INSERT INTO Person\_PhoneNo (SSN, PhoneNo)  
VALUES (301112233, 1234567908),  
(301112233, 1234567909);



-- Insert two unique phone numbers for Chris Anderson

```
INSERT INTO Person_PhoneNo (SSN, PhoneNo)
```

```
VALUES (301223344, 1234567910),
```

```
(301223344, 1234567911);
```

-- Insert two unique phone numbers for Jenny Wilson

```
INSERT INTO Person_PhoneNo (SSN, PhoneNo)
```

```
VALUES (301334455, 1234567912),
```

```
(301334455, 1234567913);
```

-- Insert two unique phone numbers for Kevin Davis

```
INSERT INTO Person_PhoneNo (SSN, PhoneNo)
```

```
VALUES (301445566, 1234567914),
```

```
(301445566, 1234567915);
```

-- Insert two unique phone numbers for Rachel Johnson

```
INSERT INTO Person_PhoneNo (SSN, PhoneNo)
```

```
VALUES (301556677, 1234567916),
```

```
(301556677, 1234567917);
```

-- Insert two unique phone numbers for Brian Lee

```
INSERT INTO Person_PhoneNo (SSN, PhoneNo)
```

```
VALUES (301667788, 1234567918),
```

```
(301667788, 1234567919);
```

-- Insert two unique phone numbers for Laura Brown

```
INSERT INTO Person_PhoneNo (SSN, PhoneNo)
```

```
VALUES (301778899, 1234567920),
```

```
(301778899, 1234567921);
```

-- Insert two unique phone numbers for Mark Miller

```
INSERT INTO Person_PhoneNo (SSN, PhoneNo)
```

```
VALUES (301889900, 1234567922),
```

```
(301889900, 1234567923);
```

-- Insert two unique phone numbers for Samantha Taylor

```
INSERT INTO Person_PhoneNo (SSN, PhoneNo)
```

```
VALUES (301990011, 1234567924),
```

```
(301990011, 1234567925);
```

-- Insert two unique phone numbers for Alex Anderson

```
INSERT INTO Person_PhoneNo (SSN, PhoneNo)
```

```
VALUES (301102030, 1234567926),
```

```
(301102030, 1234567927);
```

-- Insert two unique phone numbers for Jessica Wilson

```
INSERT INTO Person_PhoneNo (SSN, PhoneNo)
```

```
VALUES (301203040, 1234567928),
```

```
(301203040, 1234567929);
```

-- Insert two unique phone numbers for Matt Davis

```
INSERT INTO Person_PhoneNo (SSN, PhoneNo)
```

```
VALUES (301304050, 1234567930),
```

```
(301304050, 1234567931);
```



-- Insert two unique phone numbers for Olivia Johnson

```
INSERT INTO Person_PhoneNo (SSN, PhoneNo)
VALUES (301405060, 1234567932),
       (301405060, 1234567933);
```

-- Insert two unique phone numbers for Chris Lee

```
INSERT INTO Person_PhoneNo (SSN, PhoneNo)
VALUES (301506070, 1234567934),
       (301506070, 1234567935);
```

-- Insert two unique phone numbers for Nicole Brown

```
INSERT INTO Person_PhoneNo (SSN, PhoneNo)
VALUES (301607080, 1234567936),
       (301607080, 1234567937);
```

-- Insert two unique phone numbers for Adam Miller

```
INSERT INTO Person_PhoneNo (SSN, PhoneNo)
VALUES (301708090, 1234567938),
       (301708090, 1234567939);
```

-- Insert two unique phone numbers for Emily Taylor

```
INSERT INTO Person_PhoneNo (SSN, PhoneNo)
VALUES (301809010, 1234567940),
       (301809010, 1234567941);
```

-- Insert two unique phone numbers for Mike Anderson

```
INSERT INTO Person_PhoneNo (SSN, PhoneNo)
VALUES (301910203, 1234567942),
       (301910203, 1234567943);
```

-- Insert two unique phone numbers for Julia Wilson

```
INSERT INTO Person_PhoneNo (SSN, PhoneNo)
VALUES (301112244, 1234567944),
       (301112244, 1234567945);
```

-- Insert two unique phone numbers for Tommy Davis

```
INSERT INTO Person_PhoneNo (SSN, PhoneNo)
VALUES (301223346, 1234567946),
       (301223346, 1234567947);
```

-- Insert two unique phone numbers for Hannah Johnson

```
INSERT INTO Person_PhoneNo (SSN, PhoneNo)
VALUES (301334458, 1234567948),
       (301334458, 1234567949);
```

INSERT INTO Employee (SSN, EmployeeID, Salary, WorkingHours, Shift) VALUES

```
(301123456, 1001, 50000, 40, 'Morning'),
(301234567, 1002, 60000, 45, 'Afternoon'),
(301345678, 1003, 55000, 40, 'Evening'),
(301456789, 1004, 70000, 50, 'Afternoon'),
(301567890, 1005, 45000, 35, 'Morning'),
(301678901, 1006, 65000, 40, 'Afternoon'),
(301789012, 1007, 80000, 50, 'Morning'),
(301890123, 1008, 55000, 40, 'Afternoon'),
(301901234, 1009, 50000, 40, 'Morning'),
(301112233, 1010, 60000, 45, 'Afternoon'),
```



```
(301223344, 1011, 55000, 40, 'Morning'),  
(301334455, 1012, 70000, 50, 'Evening'),  
(301445566, 1013, 45000, 35, 'Morning'),  
(301556677, 1014, 65000, 40, 'Afternoon'),  
(301667788, 1015, 80000, 50, 'Morning'),  
(301778899, 1016, 55000, 40, 'Afternoon'),  
(301889900, 1017, 50000, 40, 'Morning'),  
(301990011, 1018, 60000, 45, 'Afternoon'),  
(301102030, 1019, 55000, 40, 'Morning'),  
(301203040, 1020, 70000, 50, 'Evening'),  
(301304050, '1021', 50000, 45, 'Morning'),  
(301405060, '1022', 45000, 45, 'Afternoon'),  
(301334789, '1023', 60000, 50, 'Morning');
```

```
INSERT INTO Receptionist (SSN, DeskNo)  
VALUES  
(301123456, 1),  
(301234567, 2),  
(301345678, 3),  
(301456789, 4),  
(301567890, 5);
```

```
INSERT INTO Admin (SSN, AdminID) VALUES (301678901, 2001);  
INSERT INTO Admin (SSN, AdminID) VALUES (301789012, 2002);  
INSERT INTO Admin (SSN, AdminID) VALUES (301890123, 2003);  
INSERT INTO Admin (SSN, AdminID) VALUES (301901234, 2004);  
INSERT INTO Admin (SSN, AdminID) VALUES (301112233, 2005);
```

```
INSERT INTO Department (DepartmentID, DepartmentName, MGRSSN) VALUES (101, 'Cardiology Department', NULL);  
INSERT INTO Department (DepartmentID, DepartmentName, MGRSSN) VALUES (102, 'Pediatrics Department', NULL);  
INSERT INTO Department (DepartmentID, DepartmentName, MGRSSN) VALUES (103, 'Dermatology Department', NULL);  
INSERT INTO Department (DepartmentID, DepartmentName, MGRSSN) VALUES (104, 'Orthopedics Department', NULL);  
INSERT INTO Department (DepartmentID, DepartmentName, MGRSSN) VALUES (105, 'Neurology Department', NULL);  
INSERT INTO Department (DepartmentID, DepartmentName, MGRSSN) VALUES (106, 'Emergency Department', NULL);
```

```
INSERT INTO Doctor (SSN, Speciality, DepartmentID, ClinicID) VALUES  
(301223344, 'Heart Failure and Transplant', 101, NULL),  
(301334455, 'Neonatology', 102, NULL),  
(301445566, 'Dermatopathology', 103, NULL),  
(301556677, 'Hand Surgery', 104, NULL),  
(301667788, 'Neurogenetics', 105, NULL),  
(301778899, 'Trauma Surgery', 106, NULL),  
(301334789, 'Heart Failure and Transplant', 101, NULL);
```

```
INSERT INTO Clinic (ClinicID, ClinicType, FloorNo) VALUES  
(1, 'Heart Diseases Clinic', 3),  
(2, 'Pediatric Clinic', 1),  
(3, 'Skin Diseases Clinic', 2),  
(4, 'Ortho-Organ Clinic', 2);  
Update Department  
SET MGRSSN = 301223344  
WHERE DepartmentID = 101;
```

```
Update Department  
SET MGRSSN = 301334455  
WHERE DepartmentID = 102;
```



Update Department

SET MGRSSN = 301445566

WHERE DepartmentID = 103;

Update Department

SET MGRSSN = 301556677

WHERE DepartmentID = 104;

Update Department

SET MGRSSN = 301667788

WHERE DepartmentID = 105;

Update Department

SET MGRSSN = 301778899

WHERE DepartmentID = 106;

UPDATE DOCTOR

SET ClinicID = 1

WHERE Speciality = 'Heart Failure and Transplant';

UPDATE DOCTOR

SET ClinicID = 2

WHERE Speciality = 'Neonatology';

UPDATE DOCTOR

SET ClinicID = 3

WHERE Speciality = 'Dermatopathology';

UPDATE DOCTOR

SET ClinicID = 4

WHERE Speciality = 'Hand Surgery';

INSERT INTO Nurse (SSN, NursingService, DepartmentID) VALUES

(301889900, 'Medication administration and monitoring vital signs', 101),

(301990011, 'Patient education and wound care', 102),

(301102030, 'Assisting with procedures and treatments', 103),

(301203040, 'Pain management and symptom control', 104),

(301304050, 'Neurological assessment and care planning', 105),

(301405060, 'Emergency care and triage', 106);

INSERT INTO Patient (SSN, Diagnosis, PatientID) VALUES

(301506070, 'Myocardial Infarction', 1001),

(301607080, 'Trigger Finger', 1002),

(301708090, 'Migraine', 1003),

(301809010, 'Psoriasis', 1004),

(301910203, 'Osteoarthritis', 1005),

(301112244, 'Rheumatoid Arthritis', 1006),

(301223346, 'Epilepsy', 1007),

(301334458, 'Acute Coronary Syndrom', 1008);

INSERT INTO MedicalRecord (MedicalRecordID, ExaminationDate, Description, PatientSSN) VALUES

(1, '2022-01-15', 'Patient had a routine check-up and was advised to engage in regular exercise and eat a healthy diet.', 301506070),

(2, '2022-02-01', 'Patient came in with a minor cut on their hand and received a tetanus shot.', 301607080),

(3, '2022-03-10', 'Patient had a physical exam and was found to have high blood pressure. They were prescribed medication to manage it.', 301708090),

(4, '2022-04-05', 'Patient came in with a sprained ankle and was advised to rest, ice, and elevate the affected area.', 301809010),

(5, '2022-05-20', 'Patient had a routine eye exam and was prescribed glasses for nearsightedness.', 301910203);



INSERT INTO Room (RoomID, FloorNo) VALUES

(501, 1),  
(502, 2),  
(503, 3),  
(504, 3);

INSERT INTO MedicalCareRoom (RoomID, BedsCount) VALUES

(501, 10),  
(502, 15);

INSERT INTO OperationRoom (RoomID, MaxDoctorsNo) VALUES

(503, 3),  
(504, 2);

INSERT INTO InPatient (SSN, AdmissionDate, DischargeDate, OperationID, RoomID)  
VALUES

(301506070, '2022-05-15', '2022-05-25', NULL, 501),  
(301607080, '2022-05-01', '2022-05-10', NULL, 502),  
(301708090, '2022-05-10', '2022-05-20', NULL, 501),  
(301809010, '2022-04-05', '2022-04-12', NULL, 502),  
(301910203, '2022-06-15', '2022-06-22', NULL, 501),  
(301112244, '2022-07-15', NULL, NULL, 502),  
(301223346, '2022-08-17', NULL, NULL, 501);

INSERT INTO OutPatient (SSN, ReservationNo) VALUES

(301607080, 701),  
(301334458, 702);

INSERT INTO Operation (OperationID, OperationName, OperationDate, OperationRoomID) VALUES

(901, 'Heart Transplant', '2022-05-20', 503),  
(902, 'Trigger Finger Release', '2022-05-05', 504);

UPDATE InPatient

SET OperationID = 901

WHERE SSN IN

(SELECT SSN

FROM Patient

WHERE PatientID = 1001 AND Diagnosis = 'Myocardial Infarction');

UPDATE InPatient

SET OperationID = 902

WHERE SSN IN

(SELECT SSN

FROM Patient

WHERE PatientID = 1002 AND Diagnosis = 'Trigger Finger');

-- Add equipment for OperationRoom ID 503

INSERT INTO OperationRoom\_Equipment (OperationRoomID, Equipment) VALUES

(503, 'Surgical table'),  
(503, 'Heart-lung machine'),  
(503, 'Intra-aortic balloon pump'),  
(503, 'Defibrillator'),  
(503, 'Cardiac output monitor'),  
(503, 'Transesophageal echocardiography');





```
-- Add equipment for OperationRoom ID 504
INSERT INTO OperationRoom_Equipment (OperationRoomID, Equipment) VALUES
(504, 'Surgical table'),
(504, 'Operating lights'),
(504, 'Anesthesia machine'),
(504, 'Surgical instruments'),
(504, 'Tourniquet');
```

```
INSERT INTO PERFORMS (DoctorSSN, OperationID)
VALUES (301223344, 901);
```

```
INSERT INTO PERFORMS (DoctorSSN, OperationID)
VALUES (301334789, 901);
```

```
INSERT INTO PERFORMS (DoctorSSN, OperationID)
VALUES (301556677, 902);
```

```
INSERT INTO VISITS (OutPatientSSN, ClinicID, VisitDate) VALUES
(301607080, 4, '2022-04-25');
```

```
INSERT INTO VISITS (OutPatientSSN, ClinicID, VisitDate) VALUES
(301334458, 1, '2022-09-18');
```

## • SQL Updates and Deletes

-- Insert a person and then apply update and delete

```
INSERT INTO Person (SSN, FNAME, LNAME, BirthDate, Gender, Address) VALUES
(301123477, 'Lisa', 'Reinhart', '1995-01-01', 'F', '125 Main St');
```

-- Insert two unique phone numbers for Lisa Reinhart

```
INSERT INTO Person_PhoneNo (SSN, PhoneNo)
VALUES (301123477, 1234567777);
```

```
INSERT INTO Employee (SSN, EmployeeID, Salary, WorkingHours, Shift) VALUES
(301123477, 1024, 60000, 40, 'Morning');
```

```
INSERT INTO Doctor (SSN, Speciality, DepartmentID, ClinicID) VALUES
(301123477, 'Neonatology', 102, 2);
```

```
SELECT P.SSN,P.FNAME,P.LNAME,E.EmployeeID, E.Salary, E.WorkingHours, E.Shift, D.Speciality, D.DepartmentID,D.ClinicID
FROM PERSON AS P,EMPLOYEE AS E,DOCTOR AS D
WHERE P.SSN = 301123477 AND P.SSN = E.SSN AND E.SSN = D.SSN;
```

	SSN	FNAME	LNAME	EmployeeID	Salary	WorkingHours	Shift	Speciality	DepartmentID	ClinicID
▶	301123477	Lisa	Reinhart	1024	60000	40	Morning	Neonatology	102	2



-- Update SSN of a Person

UPDATE PERSON

SET SSN = 301123479

WHERE SSN = 301123477;

SELECT P.SSN,P.FNAME,P.LNAME,E.EmployeeID, E.Salary, E.WorkingHours, E.Shift, D.Speciality, D.DepartmentID,D.ClinicID

FROM PERSON AS P,EMPLOYEE AS E,DOCTOR AS D

WHERE P.SSN = 301123479 AND P.SSN = E.SSN AND E.SSN = D.SSN;

	SSN	FNAME	LNAME	EmployeeID	Salary	WorkingHours	Shift	Speciality	DepartmentID	ClinicID
▶	301123479	Lisa	Reinhart	1024	60000	40	Morning	Neonatology	102	2

-- Update ClinicID to Null of a Doctor

UPDATE DOCTOR

SET CLINICID = NULL

WHERE SSN = 301123479;

SELECT P.SSN,P.FNAME,P.LNAME,E.EmployeeID, E.Salary, E.WorkingHours, E.Shift, D.Speciality, D.DepartmentID,D.ClinicID

FROM PERSON AS P,EMPLOYEE AS E,DOCTOR AS D

WHERE P.SSN = 301123479 AND P.SSN = E.SSN AND E.SSN = D.SSN;

	SSN	FNAME	LNAME	EmployeeID	Salary	WorkingHours	Shift	Speciality	DepartmentID	ClinicID
▶	301123479	Lisa	Reinhart	1024	60000	40	Morning	Neonatology	102	NULL

-- Update Salary of Employee

UPDATE EMPLOYEE

SET SALARY = SALARY + SALARY \* 0.1

WHERE SSN = 301123479;

SELECT P.SSN,P.FNAME,P.LNAME,E.EmployeeID, E.Salary, E.WorkingHours, E.Shift, D.Speciality, D.DepartmentID,D.ClinicID

FROM PERSON AS P,EMPLOYEE AS E,DOCTOR AS D

WHERE P.SSN = 301123479 AND P.SSN = E.SSN AND E.SSN = D.SSN;

	SSN	FNAME	LNAME	EmployeeID	Salary	WorkingHours	Shift	Speciality	DepartmentID	ClinicID
▶	301123479	Lisa	Reinhart	1024	66000	40	Morning	Neonatology	102	NULL



-- Delete Person

DELETE FROM PERSON

WHERE SSN = 301123479;

SELECT P.SSN,P.FNAME,P.LNAME,E.EmployeeID, E.Salary, E.WorkingHours, E.Shift, D.Speciality, D.DepartmentID,D.ClinicID

FROM PERSON AS P,EMPLOYEE AS E,DOCTOR AS D

WHERE P.SSN = 301123479 AND P.SSN = E.SSN AND E.SSN = D.SSN;

	SSN	FNAME	LNAME	EmployeeID	Salary	WorkingHours	Shift	Speciality	DepartmentID	ClinicID
--	-----	-------	-------	------------	--------	--------------	-------	------------	--------------	----------

- **SQL Report Retrivals**

-- Get all doctors who manage a department

SELECT P.SSN, P.FNAME, P.LNAME, D.Speciality

FROM Person AS P

INNER JOIN Doctor AS D ON P.SSN = D.SSN

WHERE D.SSN IN

(SELECT MGRSSN

FROM Department);

	SSN	FNAME	LNAME	Speciality
▶	301223344	Chris	Anderson	Heart Failure and Transplant
	301334455	Jenny	Wilson	Neonatology
	301445566	Kevin	Davis	Dermatopathology
	301556677	Rachel	Johnson	Hand Surgery
	301667788	Brian	Lee	Neurogenetics
	301778899	Laura	Brown	Trauma Surgery



-- Get count of doctors and nurses in each department

```
SELECT DEP.DepartmentID, DEP.DepartmentName,  
  
       COUNT(DISTINCT DOC.SSN) AS DoctorCount,  
  
       COUNT(DISTINCT NUR.SSN) AS NurseCount,  
  
       COUNT(DISTINCT DOC.SSN) + COUNT(DISTINCT NUR.SSN) AS  
StaffCount  
  
FROM Department AS DEP  
  
LEFT JOIN Doctor AS DOC ON DEP.DepartmentID = DOC.DepartmentID  
  
LEFT JOIN Nurse AS NUR ON DEP.DepartmentID = NUR.DepartmentID  
  
GROUP BY DEP.DepartmentID, DEP.DepartmentName;
```

	DepartmentID	DepartmentName	DoctorCount	NurseCount	StaffCount
▶	101	Cardiology Department	2	1	3
	102	Pediatrics Department	1	1	2
	103	Dermatology Department	1	1	2
	104	Orthopedics Department	1	1	2
	105	Neurology Department	1	1	2
	106	Emergency Department	1	1	2

-- Get number of all inpatients who stayed in each medical care room

```
SELECT MR.RoomID, COUNT(INP.SSN) AS NoPatients  
  
FROM MedicalCareRoom AS MR  
  
INNER JOIN InPatient AS INP ON MR.RoomID = INP.RoomID  
  
GROUP BY MR.RoomID;
```

	RoomID	NoPatients
▶	501	4
	502	3



-- Get average and total salary for each department

```
SELECT Dept.DepartmentID, Dept.DepartmentName,  
  
       AVG(Emp.Salary) AS AvgSalary,  
  
       SUM(Emp.Salary) AS TotalSalary  
  
FROM (  
  
    SELECT D.SSN, D.DepartmentID, E.Salary  
  
    FROM Doctor AS D  
  
    INNER JOIN Employee AS E ON D.SSN = E.SSN  
  
    UNION  
  
    SELECT N.SSN, N.DepartmentID, E.Salary  
  
    FROM Nurse AS N  
  
    INNER JOIN Employee AS E ON N.SSN = E.SSN  
  
    ) AS Emp  
  
INNER JOIN Department AS Dept ON Emp.DepartmentID = Dept.DepartmentID  
  
GROUP BY Dept.DepartmentID, Dept.DepartmentName;
```

	DepartmentID	DepartmentName	AvgSalary	TotalSalary
▶	101	Cardiology Department	55000.0000	165000
	102	Pediatrics Department	65000.0000	130000
	103	Dermatology Department	50000.0000	100000
	104	Orthopedics Department	67500.0000	135000
	105	Neurology Department	65000.0000	130000
	106	Emergency Department	50000.0000	100000



-- Get doctors who work in a clinic

```
SELECT P.SSN, P.FNAME, P.LNAME, D.ClinicID  
  
FROM Person AS P  
  
INNER JOIN Doctor AS D ON P.SSN = D.SSN  
  
WHERE D.ClinicID IS NOT NULL;
```

	SSN	FNAME	LNAME	ClinicID
▶	301223344	Chris	Anderson	1
	301334789	Anakin	Shephard	1
	301334455	Jenny	Wilson	2
	301445566	Kevin	Davis	3
	301556677	Rachel	Johnson	4

-- Get all doctors who have performed at least one operation

```
SELECT SSN, FNAME, LNAME  
  
FROM Person  
  
WHERE EXISTS (  
  
    SELECT * FROM Doctor  
  
    INNER JOIN PERFORMS ON Doctor.SSN = PERFORMS.DoctorSSN  
  
    WHERE Doctor.SSN = Person.SSN  
  
);
```

	SSN	FNAME	LNAME
▶	301223344	Chris	Anderson
	301334789	Anakin	Shephard
	301556677	Rachel	Johnson
•	NULL	NULL	NULL



-- Get all persons who work as receptionist

```
SELECT P.SSN, P.FNAME, P.LNAME  
  
FROM PERSON AS P, Receptionist AS R  
  
WHERE P.SSN = R.SSN;
```

	SSN	FNAME	LNAME
▶	301123456	John	Doe
	301234567	Jane	Doe
	301345678	Bob	Smith
	301456789	Sarah	Johnson
	301567890	Mike	Williams

-- Get Phone No. for every nurse

```
SELECT P.SSN, P.FNAME, P.LNAME, Ph.PhoneNo  
  
FROM Person AS P  
  
INNER JOIN Person_PhoneNo AS Ph ON P.SSN = Ph.SSN  
  
INNER JOIN Nurse AS N ON P.SSN = N.SSN;
```

	SSN	FNAME	LNAME	PhoneNo
▶	301889900	Mark	Miller	1234567922
	301889900	Mark	Miller	1234567923
	301990011	Samantha	Taylor	1234567924
	301990011	Samantha	Taylor	1234567925
	301102030	Alex	Anderson	1234567926
	301102030	Alex	Anderson	1234567927
	301203040	Jessica	Wilson	1234567928
	301203040	Jessica	Wilson	1234567929
	301304050	Matt	Davis	1234567930
	301304050	Matt	Davis	1234567931
	301405060	Olivia	Johnson	1234567932
	301405060	Olivia	Johnson	1234567933



-- Get medical record for every inpatient

```
SELECT P.SSN, P.FNAME,P.LNAME, Pt.PatientID, MR.Description,  
MR.ExaminationDate  
  
FROM PERSON AS P, Patient AS Pt, InPatient AS IPt, MedicalRecord AS MR  
  
WHERE P.SSN = Pt.SSN AND IPt.SSN = Pt.SSN AND MR.PatientSSN = IPt.SSN;
```

SSN	FNAME	LNAME	PatientID	Description	ExaminationDate
301506070	Chris	Lee	1001	Patient had a routine check-up and was advised to engage in regular exercise and eat a healthy diet.	2022-01-15
301607080	Nicole	Brown	1002	Patient came in with a minor cut on their hand and received a tetanus shot.	2022-02-01
301708090	Adam	Miller	1003	Patient had a physical exam and was found to have high blood pressure. They were prescribed medication to manage it.	2022-03-10
301809010	Emily	Taylor	1004	Patient came in with a sprained ankle and was advised to rest, ice, and elevate the affected area.	2022-04-05
301910203	Mike	Anderson	1005	Patient had a routine eye exam and was prescribed glasses for nearsightedness.	2022-05-20

-- Get count of Receptionists in hospital

```
SELECT COUNT(*) AS ReceptionistCount  
  
FROM Receptionist;
```

	ReceptionistCount
▶	5

-- Get No of Doctors in each Clinic

```
SELECT Clinic.ClinicID, Clinic.ClinicType,  
  
COUNT(DISTINCT doctor.ssn) AS NoDoctors  
  
FROM Clinic JOIN Doctor ON Doctor.ClinicID = Clinic.ClinicID  
  
GROUP BY Clinic.ClinicID, Clinic.ClinicType;
```

	ClinicID	ClinicType	NoDoctors
▶	1	Heart Diseases Clinic	2
	2	Pediatric Clinic	1
	3	Skin Diseases Clinic	1
	4	Ortho-Organ Clinic	1





-- Get the name and description of medical records associated with a patient who has a reservation at a particular Clinic

```
SELECT p.FNAME, p.LNAME, mr.Description  
  
FROM Person p  
  
JOIN Patient pat ON p.SSN = pat.SSN  
  
JOIN OutPatient op ON pat.SSN = op.SSN  
  
JOIN VISITS v ON op.SSN = v.OutPatientSSN  
  
JOIN Clinic c ON v.ClinicID = c.ClinicID  
  
JOIN MedicalRecord mr ON pat.SSN = mr.PatientSSN  
  
WHERE c.ClinicID = 4;
```

	FNAME	LNAME	Description
▶	Nicole	Brown	Patient came in with a minor cut on their hand and received a tetanus shot.

-- Get the names and admission dates of inpatients who have not yet been discharged

```
SELECT p.FNAME, p.LNAME, ip.AdmissionDate  
  
FROM Person p  
  
JOIN Patient pat ON p.SSN = pat.SSN  
  
JOIN InPatient ip ON pat.SSN = ip.SSN  
  
WHERE ip.DischargeDate IS NULL;
```

	FNAME	LNAME	AdmissionDate
▶	Julia	Wilson	2022-07-15
	Tommy	Davis	2022-08-17

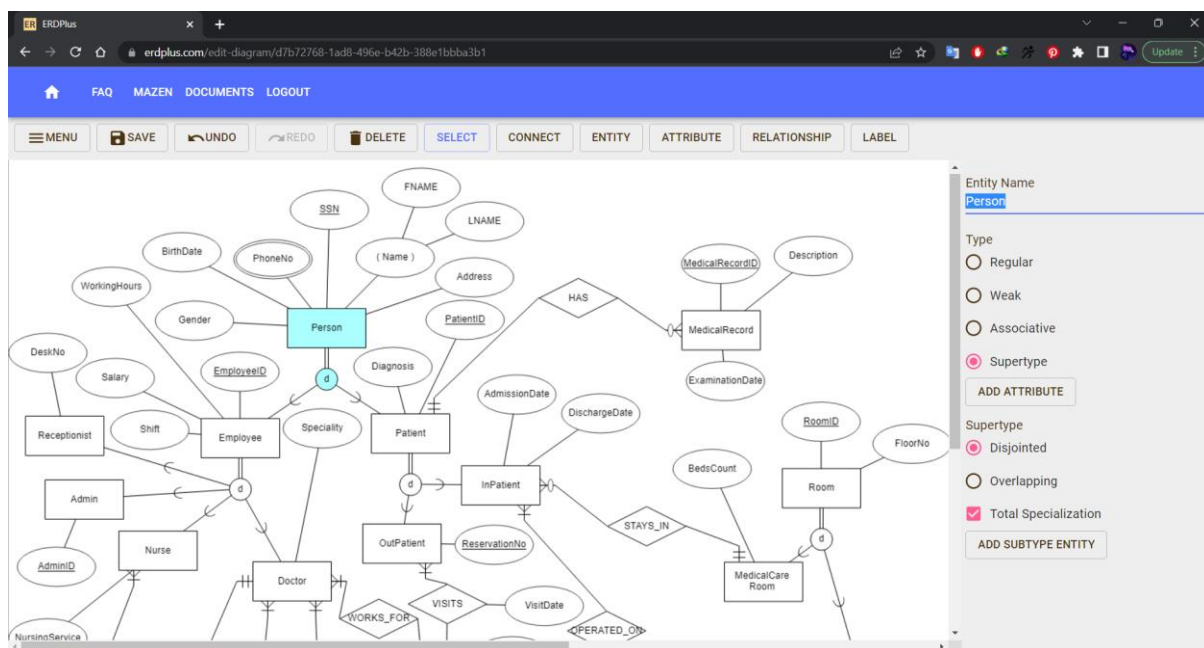
## 7. Implementation

### 7.1 Part 1: using ERD Tool

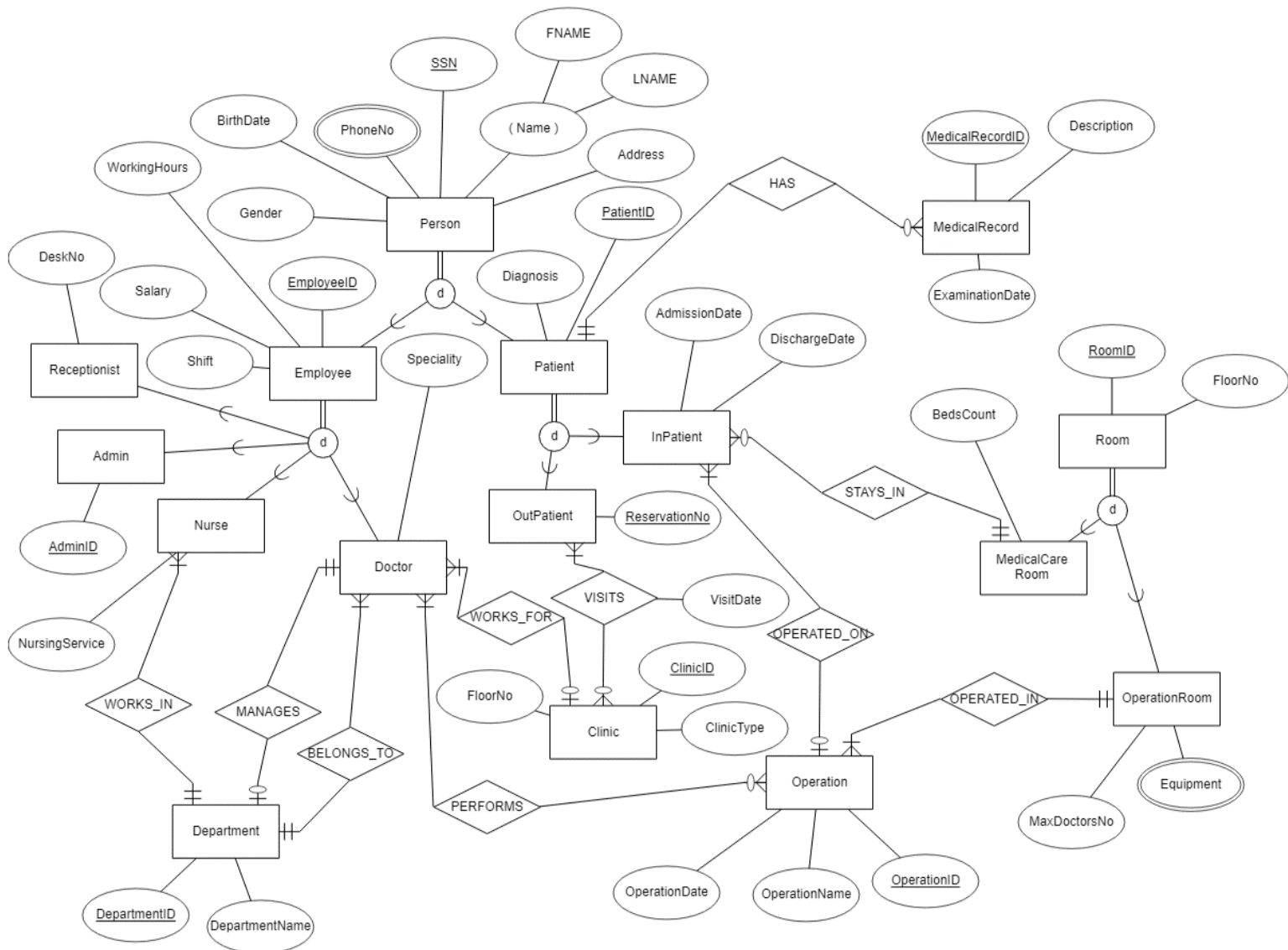
Tool Used is ERD Plus: It's a web-based database modeling tool which let you easily and quickly create your

- ER Diagrams
- Relational Schemas

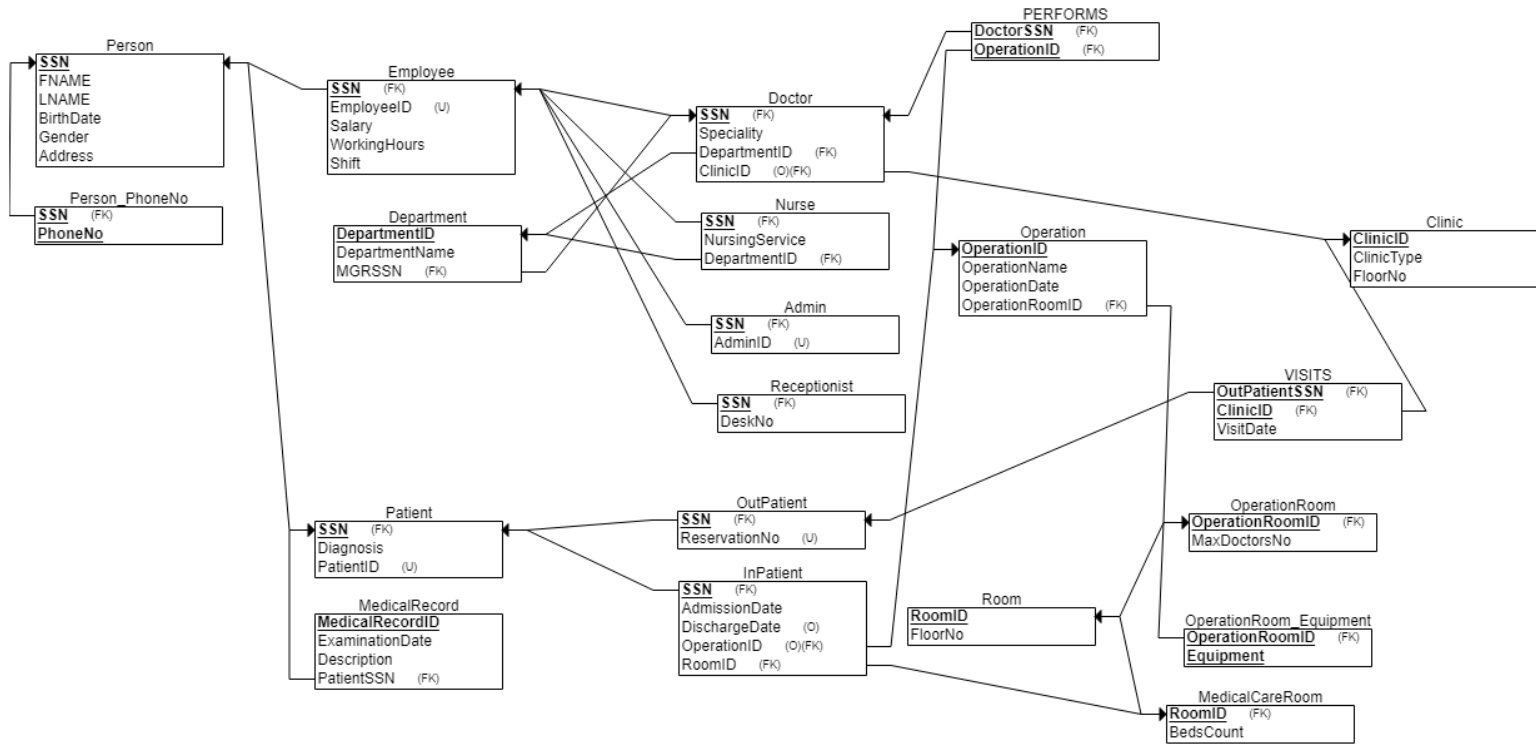
Also, you can automatically convert your ER Diagrams to Relational Schemas, Export them to images and Export SQL.



EER Diagram:



Relational Schema:





**SQL:**

```
CREATE TABLE Person
(
    SSN INT NOT NULL,
    FNAME VARCHAR(50) NOT NULL,
    LNAME VARCHAR(50) NOT NULL,
    BirthDate DATE NOT NULL,
    Gender CHAR(1) NOT NULL,
    Address VARCHAR(100) NOT NULL,
    PRIMARY KEY (SSN)
);

CREATE TABLE Patient
(
    SSN INT NOT NULL,
    Diagnosis VARCHAR(50) NOT NULL,
    PatientID INT NOT NULL,
    PRIMARY KEY (SSN),
    FOREIGN KEY (SSN) REFERENCES Person(SSN)
    ON DELETE CASCADE
    ON UPDATE CASCADE,
    UNIQUE (PatientID)
);

CREATE TABLE Employee
(
    SSN INT NOT NULL,
    EmployeeID INT NOT NULL,
    Salary INT NOT NULL,
    WorkingHours INT NOT NULL,
    Shift VARCHAR(50) NOT NULL,
    PRIMARY KEY (SSN),
    FOREIGN KEY (SSN) REFERENCES Person(SSN)
    ON DELETE CASCADE
    ON UPDATE CASCADE,
    UNIQUE (EmployeeID)
);

CREATE TABLE Room
(
    RoomID INT NOT NULL,
    FloorNo INT NOT NULL,
    PRIMARY KEY (RoomID)
);

CREATE TABLE Receptionist
(
    SSN INT NOT NULL,
    DeskNo INT NOT NULL,
    PRIMARY KEY (SSN),
    FOREIGN KEY (SSN) REFERENCES Employee(SSN)
    ON DELETE CASCADE
    ON UPDATE CASCADE
);
```



```
CREATE TABLE MedicalCareRoom
(
    RoomID INT NOT NULL,
    BedsCount INT NOT NULL,
    PRIMARY KEY (RoomID),
    FOREIGN KEY (RoomID) REFERENCES Room(RoomID)
    ON DELETE CASCADE
    ON UPDATE CASCADE
);
CREATE TABLE OperationRoom
(
    RoomID INT NOT NULL,
    MaxDoctorsNo INT NOT NULL,
    PRIMARY KEY (RoomID),
    FOREIGN KEY (RoomID) REFERENCES Room(RoomID)
    ON DELETE CASCADE
    ON UPDATE CASCADE
);
CREATE TABLE MedicalRecord
(
    MedicalRecordID INT NOT NULL,
    ExaminationDate DATE NOT NULL,
    Description VARCHAR(150) NOT NULL,
    PatientSSN INT NOT NULL,
    PRIMARY KEY (MedicalRecordID),
    FOREIGN KEY (PatientSSN) REFERENCES Patient(SSN)
    ON DELETE CASCADE
    ON UPDATE CASCADE
);
CREATE TABLE Admin
(
    SSN INT NOT NULL,
    AdminID INT NOT NULL,
    PRIMARY KEY (SSN),
    FOREIGN KEY (SSN) REFERENCES Employee(SSN)
    ON DELETE CASCADE
    ON UPDATE CASCADE,
    UNIQUE (AdminID)
);
CREATE TABLE OutPatient
(
    SSN INT NOT NULL,
    ReservationNo INT NOT NULL,
    PRIMARY KEY (SSN),
    FOREIGN KEY (SSN) REFERENCES Patient(SSN)
    ON DELETE CASCADE
    ON UPDATE CASCADE,
    UNIQUE (ReservationNo)
);
CREATE TABLE Clinic
(
    ClinicID INT NOT NULL,
    ClinicType VARCHAR(50) NOT NULL,
    FloorNo INT NOT NULL,
    PRIMARY KEY (ClinicID)
);
```



```
CREATE TABLE VISITS
(
  OutPatientSSN INT NOT NULL,
  ClinicID INT NOT NULL,
  VisitDate DATE NOT NULL,
  PRIMARY KEY (OutPatientSSN, ClinicID),
  FOREIGN KEY (OutPatientSSN) REFERENCES OutPatient(SSN)
  ON DELETE RESTRICT
  ON UPDATE CASCADE,
  FOREIGN KEY (ClinicID) REFERENCES Clinic(ClinicID)
  ON DELETE CASCADE
  ON UPDATE CASCADE
);

CREATE TABLE Person_PhoneNo
(
  SSN INT NOT NULL,
  PhoneNo INT NOT NULL,
  PRIMARY KEY (PhoneNo, SSN),
  FOREIGN KEY (SSN) REFERENCES Person(SSN)
  ON DELETE CASCADE
  ON UPDATE CASCADE
);

CREATE TABLE OperationRoom_Equipment
(
  OperationRoomID INT NOT NULL,
  Equipment VARCHAR(50) NOT NULL,
  PRIMARY KEY (Equipment, OperationRoomID),
  FOREIGN KEY (OperationRoomID) REFERENCES OperationRoom(RoomID)
  ON DELETE CASCADE
  ON UPDATE CASCADE
);

CREATE TABLE Operation
(
  OperationID INT NOT NULL,
  OperationName VARCHAR(50) NOT NULL,
  OperationDate DATE NOT NULL,
  OperationRoomID INT NOT NULL,
  PRIMARY KEY (OperationID),
  FOREIGN KEY (OperationRoomID) REFERENCES OperationRoom(RoomID)
  ON DELETE RESTRICT
  ON UPDATE CASCADE
);
```



```
CREATE TABLE InPatient
(
    SSN INT NOT NULL,
    AdmissionDate DATE NOT NULL,
    DischargeDate DATE,
    OperationID INT,
    RoomID INT NOT NULL,
    PRIMARY KEY (SSN),
    FOREIGN KEY (SSN) REFERENCES Patient(SSN)
    ON DELETE CASCADE
    ON UPDATE CASCADE,
    FOREIGN KEY (OperationID) REFERENCES Operation(OperationID)
    ON DELETE RESTRICT
    ON UPDATE CASCADE,
    FOREIGN KEY (RoomID) REFERENCES MedicalCareRoom(RoomID)
    ON DELETE RESTRICT
    ON UPDATE CASCADE
);

CREATE TABLE Department
(
    DepartmentID INT NOT NULL,
    DepartmentName VARCHAR(50) NOT NULL,
    MGRSSN INT,
    PRIMARY KEY (DepartmentID)
);

CREATE TABLE Doctor
(
    SSN INT NOT NULL,
    Speciality VARCHAR(50) NOT NULL,
    DepartmentID INT NOT NULL,
    ClinicID INT,
    PRIMARY KEY (SSN),
    FOREIGN KEY (SSN) REFERENCES Employee(SSN)
    ON DELETE CASCADE
    ON UPDATE CASCADE,
    FOREIGN KEY (DepartmentID) REFERENCES Department(DepartmentID)
    ON DELETE RESTRICT
    ON UPDATE CASCADE,
    FOREIGN KEY (ClinicID) REFERENCES Clinic(ClinicID)
    ON DELETE SET NULL
    ON UPDATE CASCADE
);

CREATE TABLE Nurse
(
    SSN INT NOT NULL,
    NursingService VARCHAR(100) NOT NULL,
    DepartmentID INT NOT NULL,
    PRIMARY KEY (SSN),
    FOREIGN KEY (SSN) REFERENCES Employee(SSN)
    ON DELETE CASCADE
    ON UPDATE CASCADE,
    FOREIGN KEY (DepartmentID) REFERENCES Department(DepartmentID)
    ON DELETE RESTRICT
    ON UPDATE CASCADE
);
```

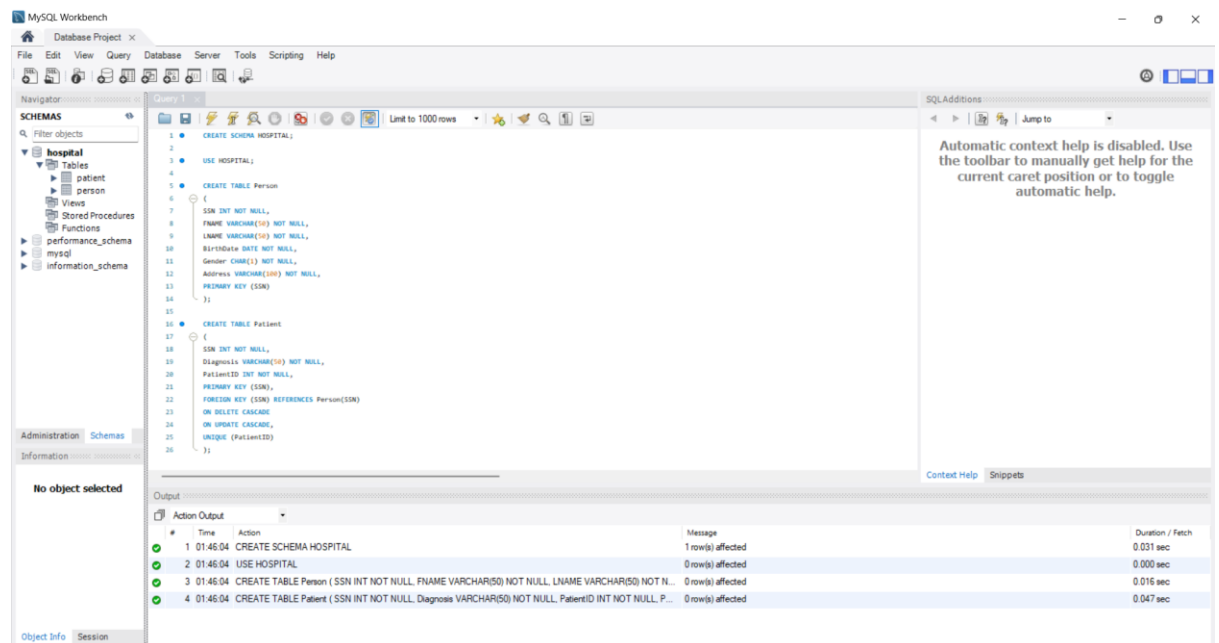


```
CREATE TABLE PERFORMS
(
  DoctorSSN INT NOT NULL,
  OperationID INT NOT NULL,
  PRIMARY KEY (DoctorSSN, OperationID),
  FOREIGN KEY (DoctorSSN) REFERENCES Doctor(SSN)
  ON DELETE RESTRICT
  ON UPDATE CASCADE,
  FOREIGN KEY (OperationID) REFERENCES Operation(OperationID)
  ON DELETE RESTRICT
  ON UPDATE CASCADE
);
```

## 7.2 Part2: using SQL Tool

Tool Used is MySQL Workbench: It's a visual database design tool that integrates SQL development, administration, database design, creation and maintenance into a single integrated development environment for the MySQL database system.

Example for Create:



### Example for Insert:

```

1 • INSERT INTO Clinic (ClinicID, ClinicType, FloorNo) VALUES
2   (1, 'Heart Diseases Clinic', 3),
3   (2, 'Pediatric Clinic', 1),
4   (3, 'Skin Diseases Clinic', 2),
5   (4, 'Ortho-Organ Clinic' , 2);
6
7 • SELECT *
8   FROM CLINIC;

```

Result Grid			
Filter Rows:			
	ClinicID	ClinicType	FloorNo
▶	1	Heart Diseases Clinic	3
	2	Pediatric Clinic	1
	3	Skin Diseases Clinic	2
	4	Ortho-Organ Clinic	2
•	NULL	NULL	NULL

### Example for Update:

```

1  -- Update Salary of Employee
2  UPDATE EMPLOYEE
3  SET SALARY = SALARY + SALARY * 0.1
4  WHERE SSN = 301123479;
5
6  SELECT P.SSN,P.FNAME,P.LNAME,E.EmployeeID, E.Salary, E.WorkingHours, E.Shift, D.Speciality, D.DepartmentID,D.ClinicID
7  FROM PERSON AS P,EMPLOYEE AS E,DOCTOR AS D
8  WHERE P.SSN = 301123479 AND P.SSN = E.SSN AND E.SSN = D.SSN;

```

Result Grid									
Filter Rows:									
	SSN	FNAME	LNAME	EmployeeID	Salary	WorkingHours	Shift	Speciality	DepartmentID
▶	301123479	Lisa	Reinhart	1024	66000	40	Morning	Neonatology	102

### Example for Delete:

```

1  -- Delete Person
2  DELETE FROM PERSON
3  WHERE SSN = 301123479;
4
5 • SELECT P.SSN,P.FNAME,P.LNAME,E.EmployeeID, E.Salary, E.WorkingHours, E.Shift, D.Speciality, D.DepartmentID,D.ClinicID
6  FROM PERSON AS P,EMPLOYEE AS E,DOCTOR AS D
7  WHERE P.SSN = 301123479 AND P.SSN = E.SSN AND E.SSN = D.SSN;

```

Result Grid									
Filter Rows:									
	SSN	FNAME	LNAME	EmployeeID	Salary	WorkingHours	Shift	Speciality	DepartmentID



Example for Select:

```
1  -- Get count of doctors and nurses in each department
2  SELECT DEP.DepartmentID, DEP.DepartmentName,
3         COUNT(DISTINCT DOC.SSN) AS DoctorCount,
4         COUNT(DISTINCT NUR.SSN) AS NurseCount,
5         COUNT(DISTINCT DOC.SSN) + COUNT(DISTINCT NUR.SSN) AS StaffCount
6  FROM Department AS DEP
7  LEFT JOIN Doctor AS DOC ON DEP.DepartmentID = DOC.DepartmentID
8  LEFT JOIN Nurse AS NUR ON DEP.DepartmentID = NUR.DepartmentID
9  GROUP BY DEP.DepartmentID, DEP.DepartmentName;
```

Result Grid

Filter Rows:

Export:

Wrap Cell Content:

	DepartmentID	DepartmentName	DoctorCount	NurseCount	StaffCount
	101	Cardiology Department	2	1	3
	102	Pediatrics Department	1	1	2
	103	Dermatology Department	1	1	2
	104	Orthopedics Department	1	1	2
	105	Neurology Department	1	1	2
	106	Emergency Department	1	1	2