```
DROP TABLE IF EXISTS CreditCardPayment CASCADE;
DROP TABLE IF EXISTS Billing CASCADE;
DROP TABLE IF EXISTS Discharge CASCADE;
DROP TABLE IF EXISTS Admission CASCADE;
DROP TABLE IF EXISTS PatientAssessment CASCADE;
DROP TABLE IF EXISTS Patient CASCADE;
DROP TABLE IF EXISTS Doctor CASCADE;
DROP TABLE IF EXISTS Nurse CASCADE;
DROP TABLE IF EXISTS Staff CASCADE;
DROP TABLE IF EXISTS Bed CASCADE;
DROP TABLE IF EXISTS Ward CASCADE;
DROP TABLE IF EXISTS Theatre CASCADE;
DROP TABLE IF EXISTS Department CASCADE;



-- Table for storing department information
CREATE TABLE Department (
    DepartmentID INT PRIMARY KEY,
    Name VARCHAR(255) NOT NULL,
    OperatingHours VARCHAR(30) NOT NULL
);

-- Table for managing ward details, including type and related department
CREATE TABLE Ward (
    WardID INT PRIMARY KEY,
    Name VARCHAR(255) NOT NULL,
    Type VARCHAR(20) CHECK (Type IN ('General', 'ICU')) NOT NULL,
    DepartmentID INT,
    FOREIGN KEY (DepartmentID) REFERENCES Department(DepartmentID)
);

-- Table to record details of beds, including dimensions and cost
CREATE TABLE Bed (
    BedID INT PRIMARY KEY,
    ComfortLevel VARCHAR(50) NOT NULL,
    Length DECIMAL(5,2) CHECK (Length > 0 AND Length <= 2.13),
    Width DECIMAL(5,2) CHECK (Width > 0 AND Width <= 1.27),
    MattressThickness DECIMAL(5,2) CHECK (MattressThickness BETWEEN 15.24 AND
17.78),
    BedCost DECIMAL(10,2) NOT NULL,
    WardID INT,
    FOREIGN KEY (WardID) REFERENCES Ward(WardID)
);

-- Table for storing information about operating theatres in various departments
CREATE TABLE Theatre (
    TheatreID INT PRIMARY KEY,
    DepartmentID INT,
    FOREIGN KEY (DepartmentID) REFERENCES Department(DepartmentID)
);

-- Table for storing staff information, including their department and salary
CREATE TABLE Staff (
    StaffID INT PRIMARY KEY,
    FullName VARCHAR(255) NOT NULL,
    Mobile VARCHAR(30) NOT NULL,
    Address TEXT NOT NULL,
    Salary DECIMAL(10,2) CHECK (Salary > 0),
```

```
    DepartmentID INT,
    FOREIGN KEY (DepartmentID) REFERENCES Department(DepartmentID)
);

-- Table for recording nurses' data, including their working clearance status
CREATE TABLE Nurse (
    StaffID INT PRIMARY KEY,
    WWCCExpiryDate DATE NOT NULL,
    FOREIGN KEY (StaffID) REFERENCES Staff(StaffID)
);

-- Table for recording doctors' information, including their specialization and
training
CREATE TABLE Doctor (
    StaffID INT PRIMARY KEY,
    Specialty VARCHAR(255) NOT NULL,
    TrainingDate DATE NOT NULL,
    ProficiencyLevel VARCHAR(20) CHECK (ProficiencyLevel IN ('Beginner',
'Intermediate', 'Advanced')) NOT NULL,
    FOREIGN KEY (StaffID) REFERENCES Staff(StaffID)
);

-- Table for managing patient records, including personal and contact details
CREATE TABLE Patient (
    PatientID INT PRIMARY KEY,
    FullName VARCHAR(255) NOT NULL,
    Email VARCHAR(255) NOT NULL UNIQUE,
    Address TEXT NOT NULL,
    DateOfBirth DATE NOT NULL,
    Mobile VARCHAR(30) NOT NULL,
    EmergencyContactName VARCHAR(255) NOT NULL,
    EmergencyContactPhone VARCHAR(15) NOT NULL,
    InsuranceNumber VARCHAR(30) NOT NULL
);

-- Table for recording patient assessments, including injury severity and time of
assessment
CREATE TABLE PatientAssessment (
    AssessmentID INT PRIMARY KEY,
    PatientID INT,
    AssessmentDate DATE NOT NULL,
    AssessmentTime TIME NOT NULL,
    InjurySeverity VARCHAR(20) CHECK (InjurySeverity IN ('Minor injury', 'Moderate
injury', 'Serious injury')) NOT NULL,
    FOREIGN KEY (PatientID) REFERENCES Patient(PatientID)
);

-- Table for tracking patient admissions, including doctor, nurse, and admission
type
CREATE TABLE Admission (
    AdmissionID INT PRIMARY KEY,
    PatientID INT,
    AdmissionDate DATE NOT NULL,
    AdmissionTime TIME NOT NULL,
    NurseID INT, -- NurseID is actually the StaffID, which is connected with the
Staff of the Staff table through the foreign key, in order to facilitate the
clearer identification of nurses or staff of other hospitals when inserting
    DoctorID INT, -- DoctorID is actually the StaffID, which is connected with the
Staff of the Staff table through the foreign key, in order to facilitate the
```

```
clearer identification of Doctors or staff of other hospitals when inserting
    ReferringPractitioner VARCHAR(255),
    ReferenceNumber VARCHAR(30),
    AdmissionType VARCHAR(20) CHECK (AdmissionType IN ('Emergency', 'Planned')) NOT
NULL,
    InjurySeverity VARCHAR(20) CHECK (InjurySeverity IN ('Minor injury', 'Moderate
injury', 'Serious injury')),
    FOREIGN KEY (PatientID) REFERENCES Patient(PatientID),
    FOREIGN KEY (NurseID) REFERENCES Nurse(StaffID),
    FOREIGN KEY (DoctorID) REFERENCES Doctor(StaffID)
);

-- Table for storing discharge details of patients, referencing their admission
record
CREATE TABLE Discharge (
    DischargeID INT PRIMARY KEY,
    AdmissionID INT,
    DischargeDate DATE NOT NULL,
    FOREIGN KEY (AdmissionID) REFERENCES Admission(AdmissionID)
);

-- Table for billing information, linked to discharge records
CREATE TABLE Billing (
    BillingID INT PRIMARY KEY,
    DischargeID INT,
    TotalCost DECIMAL(10,2) CHECK (TotalCost > 0 AND TotalCost <= 50000),
    InsuranceCoverage DECIMAL(10,2) CHECK (InsuranceCoverage >= 0),
    RemainingBalance DECIMAL(10,2) CHECK (RemainingBalance >= 0),
    InvoiceDate DATE NOT NULL,
    FOREIGN KEY (DischargeID) REFERENCES Discharge(DischargeID)
);

-- Table for recording credit card payments made by patients, linked to their
billing records
CREATE TABLE CreditCardPayment (
    CreditCardPaymentID INT PRIMARY KEY,
    BillingID INT,
    CardNumber VARCHAR(16) NOT NULL,
    ExpiryDate DATE NOT NULL,
    CardholderName VARCHAR(255) NOT NULL,
    CVV CHAR(3) NOT NULL,
    FOREIGN KEY (BillingID) REFERENCES Billing(BillingID)
);

-- Insert sample data into the Department table
INSERT INTO Department (DepartmentID, Name, OperatingHours) VALUES
(1, 'General', '10am to 8pm'),
(2, 'Emergency', '24 hours'),
(3, 'Pediatrics', '10am to 8pm'),
(4, 'Surgery', '10am to 8pm');

-- Insert sample data into the Ward table
INSERT INTO Ward (WardID, Name, Type, DepartmentID) VALUES
(1, 'Ward 1', 'General', 1),
(2, 'ICU 1', 'ICU', 3),
(3, 'Ward 4', 'General', 3),
(4, 'ICU 5', 'ICU', 4);

-- Insert sample data into the Bed table
```

```
INSERT INTO Bed (BedID, ComfortLevel, Length, Width, MattressThickness, BedCost,
WardID) VALUES
(1, 'High', 2.13, 1.27, 17.78, 123.45, 1),
(2, 'Medium', 2.00, 1.10, 16.77, 100.00, 2),
(3, 'Low', 1.80, 0.99, 15.24, 78.90, 3);

-- Insert sample data into the Theatre table
INSERT INTO Theatre (TheatreID, DepartmentID) VALUES
(1, 3),
(2, 3),
(3, 3),
(4, 3),
(5, 3),
(6, 4),
(7, 4),
(8, 4),
(9, 4),
(10, 4),
(11, 4),
(12, 4);

-- Insert sample data into the Staff table
INSERT INTO Staff (StaffID, FullName, Mobile, Address, Salary, DepartmentID) VALUES
(1, 'Nurse_1', '061400000001', 'Address_1', 80000.00, 1),
(2, 'Nurse_2', '061400000002', 'Address_2', 80000.00, 1),
(3, 'Nurse_3', '061400000003', 'Address_3', 80000.00, 1),
(4, 'Nurse_4', '061400000004', 'Address_4', 80000.00, 1),
(5, 'Nurse_5', '061400000005', 'Address_5', 80000.00, 1),
(6, 'Doctor_1', '061400000006', 'Address_6', 120000.00, 3),
(7, 'Doctor_2', '061400000007', 'Address_7', 120000.00, 3),
(8, 'Doctor_3', '061400000008', 'Address_8', 120000.00, 3),
(9, 'Doctor_4', '061400000009', 'Address_9', 120000.00, 3),
(10, 'Doctor_5', '061400000010', 'Address_10', 120000.00, 3);

-- Insert sample data into the Nurse table
INSERT INTO Nurse (StaffID, WWCCExpiryDate) VALUES
(1, '2027-09-01'),
(2, '2022-09-01'),
(3, '2023-09-01'),
(4, '2025-09-01'),
(5, '2026-09-01');

-- Insert sample data into the Doctor table
INSERT INTO Doctor (StaffID, Specialty, TrainingDate, ProficiencyLevel) VALUES
(6, 'aaaaaaaaaa', '2024-09-01', 'Advanced'),
(7, 'bbbbbbbbbb', '2024-09-01', 'Intermediate'),
(8, 'cccccccccc', '2024-09-01', 'Beginner'),
(9, 'dddddddddd', '2024-09-01', 'Advanced'),
(10, 'eeeeeeeee', '2024-09-01', 'Intermediate');

-- Insert sample data into the Patient table
INSERT INTO Patient (PatientID, FullName, Email, Address, DateOfBirth, Mobile,
EmergencyContactName, EmergencyContactPhone, InsuranceNumber) VALUES
(1, 'Patient_A', 'Patient_A@gmail.com', 'Sydney', '2000-11-11', '0614885903759',
'Father_name', '0614885903758', 'Insurance123456'),
(2, 'Patient_B', 'Patient_B@gmail.com', 'Melbourne', '2005-06-15', '061420483905',
'Mother_name', '061420483909', 'Insurance654321'),
(3, 'Patient_C', 'Patient_C@gmail.com', 'Brisbane', '1998-12-22', '061433221234',
'GrandFather_name', '061433221235', 'Insurance789012'),
```

```
(4, 'Patient_D', 'Patient_D@gmail.com', 'Perth', '1985-03-20', '061444332345',
'Friend_name', '061444332346', 'Insurance345678'),
(5, 'Patient_E', 'Patient_E@gmail.com', 'Adelaide', '2013-08-19', '061455443456',
'Colleague_name', '061455443457', 'Insurance901234');

-- Insert sample data into the PatientAssessment table
INSERT INTO PatientAssessment (AssessmentID, PatientID, AssessmentDate,
AssessmentTime, InjurySeverity) VALUES
(1, 1, '2024-09-08', '09:30:00', 'Minor injury'),
(2, 2, '2024-09-08', '09:30:00', 'Moderate injury'),
(3, 3, '2024-09-08', '09:30:00', 'Serious injury'),
(4, 4, '2024-09-08', '09:30:00', 'Moderate injury'),
(5, 5, '2024-09-08', '09:30:00', 'Minor injury');

-- Insert sample data into the Admission table
INSERT INTO Admission (AdmissionID, PatientID, AdmissionDate, AdmissionTime,
NurseID, DoctorID, ReferringPractitioner, ReferenceNumber, AdmissionType,
InjurySeverity) VALUES
(1, 1, '2024-09-08', '10:00:00', 1, 6, 'PractitionerX', 'REF0001', 'Emergency',
'Minor injury'),
(2, 2, '2024-09-08', '10:00:00', 2, 7, 'PractitionerY', 'REF0002', 'Planned',
'Moderate injury'),
(3, 3, '2024-09-08', '10:00:00', 3, 8, 'PractitionerZ', 'REF0003', 'Emergency',
'Serious injury'),
(4, 4, '2024-09-08', '10:00:00', 4, 9, 'PractitionerA', 'REF0004', 'Planned',
'Moderate injury'),
(5, 5, '2024-09-08', '10:00:00', 5, 10, 'PractitionerB', 'REF0005', 'Planned',
'Minor injury');

-- Insert sample data into the Discharge table
INSERT INTO Discharge (DischargeID, AdmissionID, DischargeDate) VALUES
(101, 1, '2024-09-09'),
(102, 2, '2024-09-09'),
(103, 3, '2024-09-09'),
(104, 4, '2024-09-09'),
(105, 5, '2024-09-09');

-- Insert sample data into the Billing table
INSERT INTO Billing (BillingID, DischargeID, TotalCost, InsuranceCoverage,
RemainingBalance, InvoiceDate) VALUES
(1, 101, 1500.00, 1000.00, 500.00, '2024-09-10'),
(2, 102, 2000.00, 1500.00, 500.00, '2024-09-10'),
(3, 103, 2500.00, 2000.00, 500.00, '2024-09-10'),
(4, 104, 3000.00, 2500.00, 500.00, '2024-09-10'),
(5, 105, 3500.00, 3000.00, 500.00, '2024-09-10');

-- Insert sample data into the CreditCardPayment table
INSERT INTO CreditCardPayment (CreditCardPaymentID, BillingID, CardNumber,
ExpiryDate, CardholderName, CVV) VALUES
(1, 1, '5211111111111111', '2026-09-09', 'holderName_1', '123'),
(2, 2, '5222222222222222', '2025-09-09', 'holderName_2', '456'),
(3, 3, '5233333333333333', '2024-09-09', 'holderName_3', '789'),
(4, 4, '5244444444444444', '2023-09-09', 'holderName_4', '012'),
(5, 5, '5255555555555555', '2027-09-09', 'holderName_5', '345');

-- Check if a nurse's WWCC certificate has expired or is still valid
SELECT
    Nurse.StaffID,
    Staff.FullName AS NurseName,
```

```
    Nurse.WWCCExpiryDate,
    CASE
        WHEN Nurse.WWCCExpiryDate < CURRENT_DATE THEN 'Expired'
        ELSE 'Valid'
    END AS Status
FROM
    Nurse
JOIN
    Staff ON Nurse.StaffID = Staff.StaffID;

-- Check if a credit card has expired or is still valid
SELECT
    CreditCardPayment.CreditCardPaymentID,
    CreditCardPayment.CardNumber,
    CreditCardPayment.ExpiryDate,
    CASE
        WHEN CreditCardPayment.ExpiryDate < CURRENT_DATE THEN 'Expired'
        ELSE 'Valid'
    END AS Status
FROM
    CreditCardPayment;


-- Get patient admissions and sort them by how serious their injuries are
SELECT
    Admission.AdmissionID,
    Patient.FullName AS PatientName,
    Admission.AdmissionDate,
    Admission.AdmissionTime,
    Admission.InjurySeverity
FROM
    Admission
JOIN
    Patient ON Admission.PatientID = Patient.PatientID
ORDER BY
    CASE Admission.InjurySeverity
        WHEN 'Serious injury' THEN 1
        WHEN 'Moderate injury' THEN 2
        WHEN 'Minor injury' THEN 3
    END;
```