



Bases de Datos I

Universidad Tecnológica Nacional - FRRO

Giuliano Crenna

Sistema de Gestión Hospitalaria

Se te encarga desarrollar un sistema para la gestión de un hospital que permita manejar la información de pacientes, doctores, citas, diagnósticos y tratamientos. El sistema debe cumplir con los siguientes requerimientos:

1. Creación de Tablas y Relaciones:

- Crear las tablas necesarias para gestionar la información de los pacientes, doctores, citas, diagnósticos y tratamientos.
- Definir las relaciones entre las tablas (llaves foráneas).

2. Inserción de Datos:

- Insertar datos de ejemplo para las tablas creadas.

3. Consultas Avanzadas:

- Realizar consultas que incluyan `JOINS` y subconsultas.
- Crear una consulta que obtenga información detallada de citas y tratamientos.

4. Procedimientos Almacenados (Stored Procedures):

- Crear procedimientos almacenados que permitan realizar operaciones de inserción y actualización en la base de datos.

5. Consultas Desafiantes:

- Realizar consultas adicionales como eliminar citas de un paciente y obtener citas sin diagnóstico.
-

1. Creación de Tablas y Relaciones

1.1 Crear una tabla para almacenar los pacientes del hospital:

1.2 Crear la tabla de doctores y relacionar con la tabla de citas:

1.3 Crear la tabla de citas que almacena las visitas de los pacientes a los doctores:

1.4 Crear la tabla de diagnósticos, relacionada con las citas:

1.5 Crear la tabla de tratamientos, relacionada con los diagnósticos:

2. Inserción de Datos

2.1 Insertar algunos pacientes:

2.2 Insertar algunos doctores:

2.3 Insertar citas entre los pacientes y los doctores:

2.4 Insertar diagnósticos realizados a los pacientes:

2.5 Insertar tratamientos asociados a los diagnósticos:

3. Consultas Avanzadas

3.1 Obtener la información completa de todos los pacientes, incluyendo el doctor y la especialidad de la cita:

3.2 Obtener el diagnóstico y tratamiento de un paciente en una fecha específica:

4. Store Procedure

4.1 Crear un **store procedure** que permita insertar nuevas citas:

4.2 Crear un **store procedure** para actualizar la información de un paciente:

5. Consultas Desafiantes

5.1 Crear un **store procedure** que elimine todas las citas de un paciente específico:

5.2 Obtener las citas que no tienen un diagnóstico asociado:

Resolución:

```
CREATE TABLE Pacientes (  
    PacienteID INT PRIMARY KEY AUTO_INCREMENT,  
    Nombre VARCHAR(50),  
    Apellido VARCHAR(50),  
    FechaNacimiento DATE,  
    Genero ENUM('M', 'F'),  
    Direccion VARCHAR(100),  
    Telefono VARCHAR(15),  
    Email VARCHAR(50)  
);  
  
CREATE TABLE Doctores (  
    DoctorID INT PRIMARY KEY AUTO_INCREMENT,  
    Nombre VARCHAR(50),  
    Apellido VARCHAR(50),  
    Especialidad VARCHAR(100),  
    Telefono VARCHAR(15),  
    Email VARCHAR(50)  
);  
  
CREATE TABLE Citas (  
    CitaID INT PRIMARY KEY AUTO_INCREMENT,  
    PacienteID INT,  
    DoctorID INT,  
    FechaCita DATETIME,  
    Motivo VARCHAR(255),  
    FOREIGN KEY (PacienteID) REFERENCES Pacientes(PacienteID),  
    FOREIGN KEY (DoctorID) REFERENCES Doctores(DoctorID)  
);  
  
CREATE TABLE Diagnosticos (  
    DiagnosticoID INT PRIMARY KEY AUTO_INCREMENT,  
    CitaID INT,  
    Descripcion TEXT,  
    FechaDiagnostico DATE,  
    FOREIGN KEY (CitaID) REFERENCES Citas(CitaID)  
);  
  
CREATE TABLE Tratamientos (  
    TratamientoID INT PRIMARY KEY AUTO_INCREMENT,  
    DiagnosticoID INT,  
    Descripcion TEXT,  
    FechaInicio DATE,  
    FechaFin DATE,
```

```

FOREIGN KEY (DiagnosticoID) REFERENCES Diagnosticos(DiagnosticoID)
);

INSERT INTO Pacientes (Nombre, Apellido, FechaNacimiento, Genero, Direccion, Telefono, Email) \
('Juan', 'Perez', '1985-05-15', 'M', 'Av. Siempre Viva 123', '123456789', 'juan.perez@email.com') \
('Maria', 'Gomez', '1990-07-20', 'F', 'Calle Falsa 456', '987654321', 'maria.gomez@email.com').

INSERT INTO Doctores (Nombre, Apellido, Especialidad, Telefono, Email) VALUES
('Luis', 'Martinez', 'Cardiología', '111222333', 'luis.martinez@hospital.com'),
('Ana', 'Ramirez', 'Neurología', '444555666', 'ana.ramirez@hospital.com');

INSERT INTO Citas (PacienteID, DoctorID, FechaCita, Motivo) VALUES
(1, 1, '2024-10-15 10:00:00', 'Chequeo cardiaco'),
(2, 2, '2024-10-16 14:00:00', 'Dolor de cabeza');

INSERT INTO Diagnosticos (CitaID, Descripcion, FechaDiagnostico) VALUES
(1, 'Hipertensión', '2024-10-15'),
(2, 'Migraña', '2024-10-16');

INSERT INTO Tratamientos (DiagnosticoID, Descripcion, FechaInicio, FechaFin) VALUES
(1, 'Medicamento para presión arterial', '2024-10-16', '2024-12-16'),
(2, 'Tratamiento para migraña', '2024-10-17', '2024-11-17');

SELECT p.Nombre AS Paciente, p.Apellido, d.Nombre AS Doctor, d.Especialidad, c.FechaCita
FROM Citas c
JOIN Pacientes p ON c.PacienteID = p.PacienteID
JOIN Doctores d ON c.DoctorID = d.DoctorID;

SELECT p.Nombre,
       p.Apellido,
       d.Descripcion AS Diagnostico,
       t.Descripcion AS Tratamiento,
       t.FechaInicio, t.FechaFin
FROM Pacientes p
JOIN Citas c ON p.PacienteID = c.PacienteID
JOIN Diagnosticos d ON c.CitaID = d.CitaID
JOIN Tratamientos t ON d.DiagnosticoID = t.DiagnosticoID
WHERE c.FechaCita = '2024-10-15';

DELIMITER //
CREATE PROCEDURE InsertarCita(
    IN p_PacienteID INT,
    IN p_DoctorID INT,
    IN p_FechaCita DATETIME,
    IN p_Motivo VARCHAR(255)
)

```

```

BEGIN
    INSERT INTO Citas (PacienteID, DoctorID, FechaCita, Motivo)
    VALUES (p_PacienteID, p_DoctorID, p_FechaCita, p_Motivo);
END //

DELIMITER //
CREATE PROCEDURE ActualizarPaciente(
    IN p_PacienteID INT,
    IN p_Nombre VARCHAR(50),
    IN p_Apellido VARCHAR(50),
    IN p_FechaNacimiento DATE,
    IN p_Genero ENUM('M', 'F'),
    IN p_Direccion VARCHAR(100),
    IN p_Telefono VARCHAR(15),
    IN p_Email VARCHAR(50)
)

BEGIN
    UPDATE Pacientes
    SET Nombre = p_Nombre, Apellido = p_Apellido, FechaNacimiento = p_FechaNacimiento,
        Genero = p_Genero, Direccion = p_Direccion, Telefono = p_Telefono, Email = p_Email
    WHERE PacienteID = p_PacienteID;
END //

DELIMITER //
CREATE PROCEDURE EliminarCitasDePaciente(
    IN p_PacienteID INT
)
BEGIN
    DELETE FROM Citas WHERE PacienteID = p_PacienteID;
END //

SELECT c.CitaID, c.FechaCita, p.Nombre, p.Apellido
FROM Citas c
LEFT JOIN Diagnosticos d ON c.CitaID = d.CitaID
JOIN Pacientes p ON c.PacienteID = p.PacienteID
WHERE d.DiagnosticoID IS NULL;

```