RETO 1

Enunciado:

La clínica La princesa necesita llevar un control informatizado de su gestión de pacientes y médicos.

De cada paciente se desea guardar el código, nombre, apellidos, dirección, población, provincia, código postal, teléfono y fecha de nacimiento.

De cada médico se desea guardar el código, nombre, apellidos, teléfono y especialidad. Se desea llevar el control de cada uno de los ingresos que el paciente hace en el hospital.

Cada ingreso que realiza el paciente queda registrado en la base de datos.

De cada ingreso se guarda el código de ingreso, el número de habitación y cama en la que el paciente realiza el ingreso y la fecha de ingreso.

Un médico puede atender varios ingresos, pero el ingreso de un paciente solo puede ser atendido por un único médico.

Un paciente puede realizar varios ingresos en el hospital.

1.- Realizar el diagrama entidad-relación.

|  |  |
| --- | --- |
| **Entidad** | **Atributo** |
| Paciente | * Código * Nombre * Apellidos * Dirección * Población * Provincia * Código Postal * Teléfono * Fecha de Nacimiento |
| Médico | * Código * Nombre * Apellido * Teléfono * Especialidad |
| Ingreso | * Código de Ingreso * Número de habitación * Cama * Fecha de ingreso |

RESTRICCIONES

* Un médico puede atender varios ingresos, pero el ingreso de un paciente solo puede ser atendido por un único médico.
* Un paciente puede realizar varios ingresos en el hospital.

RELACIONES

* Un paciente puede tener un ingreso o varios (1,N)
* Pero un ingreso pertenece a un solo paciente (1,1)
* Un médico puede atender a 1 o varios ingresos (1.N)
* Pero el ingreso de un paciente solo puede ser atendido por un único médico. (1,1)

2.- Generar los comandos DDL de creación de las tablas.

CREATE TABLE "Pacientes" (

"id\_paciente" INTEGER,

"nombre\_paciente" TEXT,

"apellidos\_paciente" TEXT,

"direccion" TEXT,

"poblacion" TEXT,

"provincia" TEXT,

"codigo\_postal" TEXT,

"telefono" INTEGER,

"fecha\_nacimiento" TEXT,

PRIMARY KEY("id\_paciente" AUTOINCREMENT)

);

CREATE TABLE "Medicos" (

"id\_medico" INTEGER,

"nombre\_medico" TEXT,

"apellidos\_medico" TEXT,

"telefono" INTEGER,

"especialidad" TEXT,

PRIMARY KEY("id\_medico" AUTOINCREMENT)

);

CREATE TABLE "Ingresos" (

"id\_ingreso" INTEGER,

"habitacion" INTEGER,

"cama" INTEGER,

"fecha\_ingreso" TEXT,

"id\_medico" INTEGER,

"id\_paciente" INTEGER,

PRIMARY KEY("id\_ingreso" AUTOINCREMENT),

FOREIGN KEY("id\_medico") REFERENCES "Medicos"("id\_medico"),

FOREIGN KEY("id\_paciente") REFERENCES "Pacientes"("id\_paciente")

);

SELECT \* FROM Pacientes

SELECT \* FROM Medicos

SELECT \* FROM Ingresosç

INSERT INTO Pacientes ( nombre\_paciente, apellidos\_paciente, direccion, poblacion, provincia, codigo\_postal, telefono, fecha\_nacimiento)

VALUES

('Alberto', 'Ruiz Muñoz', 'Calle Mayor 123', 'Madrid', 'Madrid', '28001', '912345678', '1985-03-21'),

('Bárbara', 'Calvente Domínguez', 'Av. Sol 45', 'Valencia', 'Valencia', '46001', '963852741', '1990-07-12'),

('Cinthia', 'Torres Sánchez', 'C/ Olmo 22', 'Barcelona', 'Barcelona', '08002', '934567890', '1978-11-03'),

('Daniel', 'Pérez Díaz', 'Paseo Verde 67', 'Sevilla', 'Sevilla', '41012', '955123456', '2001-06-15'),

('Ana', 'Muriel Serrano', 'Plaza del Sol 89', 'Málaga', 'Málaga', '29001', '951234567', '1988-02-08'),

('Niko', 'Nacino Muñoz', 'C/ Felix de Azara 6', 'Arroyo de la Encomienda', 'Valladolid', '47195', '692831481', '1989-10-31');

INSERT INTO Medicos (nombre\_medico, apellidos\_medico, telefono, especialidad)

VALUES

('Laura', 'García Torres', '914567890', 'Traumatología'),

('Pedro', 'Sánchez Díaz', '912345876', 'Pediatría'),

('María', 'López García', '934567123', 'Neurología'),

('José', 'Rodríguez Pérez', '955678901', 'Dermatología');

INSERT INTO Ingresos (habitacion, cama, fecha\_ingreso, id\_medico, id\_paciente)

VALUES

('101', '1', '2024-11-01', 1, 1),

('402', '2', '2024-11-15', 2, 2),

('206', '1', '2024-11-20', 3, 3),

('305', '2', '2024-11-25', 4, 4),

('405', '2', '2024-11-25', 1, 5),

('204', '1', '2024-11-20', 2, 6),

('105', '2', '2024-11-25', 3, 1);

3.- Generar las siguientes vistas:

1. Vista que muestre la información de los pacientes junto con el nombre completo del médico que los atiende.

CREATE VIEW vista\_pacientes AS

SELECT Pacientes.id\_paciente, Pacientes.nombre\_paciente, Pacientes.apellidos\_paciente, Medicos.id\_medico, Medicos.nombre\_medico, Medicos.apellidos\_medico

FROM Pacientes

INNER JOIN Ingresos

ON Pacientes.id\_paciente = Ingresos.id\_paciente

INNER JOIN Medicos

ON Ingresos.id\_medico = Medicos.id\_medico

;

SELECT \* FROM vista\_pacientes;

1. Vista que muestra la cantidad de ingresos realizados por cada paciente.

CREATE VIEW cantidad\_ingresos\_pacientes AS

SELECT p.id\_paciente, p.nombre\_paciente, p.apellidos\_paciente,

count (i.id\_ingreso) AS cantidad\_ingresos

FROM Pacientes p

LEFT JOIN Ingresos i ON p.id\_paciente = i.id\_paciente

GROUP BY p.id\_paciente, p.nombre\_paciente, p.apellidos\_paciente;

SELECT \* FROM cantidad\_ingresos\_pacientes;

1. Vista que muestre la lista de médicos junto con la cantidad de pacientes que han atendido.

CREATE VIEW cantidad\_pacientes\_atendidos AS

SELECT m.id\_medico, m.nombre\_medico, m.apellidos\_medico,

count (DISTINCT i.id\_ingreso) AS cantidad\_pacientes\_atendidos

FROM Medicos m

LEFT JOIN Ingresos i ON m.id\_medico = i.id\_medico

GROUP BY m.id\_medico, m.nombre\_medico, m.apellidos\_medico;

SELECT \* FROM cantidad\_pacientes\_atendidos

4.- Mostrar los comandos para consultar las vistas.

SELECT \* FROM vista\_pacientes;

SELECT \* FROM cantidad\_ingresos\_pacientes;

SELECT \* FROM cantidad\_pacientes\_atendidos;