

Ejercicios de SQL

Practica de tablas y consultas básicas

Ejercicio 1

1. Crear las tablas:

```
CREATE TABLE clientes (  
    dni INT AUTO_INCREMENT PRIMARY KEY,  
    nombre VARCHAR(25),  
    ciudad VARCHAR(15),  
    observaciones VARCHAR(50),  
    fecha_carnet DATE  
);  
  
CREATE TABLE coches (  
    matricula INT AUTO_INCREMENT PRIMARY KEY,  
    marca VARCHAR(25),  
    modelo VARCHAR(25),  
    color VARCHAR(15)  
);  
  
CREATE TABLE alquiler (  
    cod_alquiler INT AUTO_INCREMENT PRIMARY KEY,  
    dni_cliente INT,  
    matricula_coche INT,  
    fecha_inicio DATE,  
    fecha_entrega DATE,  
    dias_limite INT,  
    precio_alquiler DECIMAL(10, 2),  
);
```

2. Agregar columna `fecha_carnet` a la tabla clientes:

```
ALTER TABLE Clientes ADD fecha_carnet DATE;
```

3. Modificar la tabla alquiler para introducir claves ajenas.

```
ALTER TABLE alquiler  
ADD CONSTRAINT fk_dni_cliente  
FOREIGN KEY (dni_cliente)  
REFERENCES clientes(dni);  
  
ALTER TABLE alquiler
```

```
ADD CONSTRAINT fk_matricula_coche
FOREIGN KEY (matricula_coche)
REFERENCES coches(matricula);
```

4. Adjuntar la imagen correspondiente a las tablas junto con las relaciones en el documento de texto.

5. Obtener las siguientes consultas:

5.1. Todos los datos de la tabla Alquiler:

```
SELECT * FROM Alquiler;
```

5.2. Nombre y ciudad de los clientes cuyo nombre contenga una 'f' y la ciudad contenga en la segunda letra una 'm':

```
SELECT nombre, ciudad FROM Clientes
WHERE nombre LIKE '%f%' AND SUBSTRING(ciudad, 2, 1) = 'm';
```

5.3. Marca y color de los coches cuyo precio de alquiler esté entre 200 y 500 euros (ambos inclusive), ordenados por el color de forma descendente:

```
SELECT marca, color FROM Coches
WHERE precio_alquiler BETWEEN 200.00 AND 500.00
ORDER BY color DESC;
```

A continuación, se presenta la solución a los ejercicios propuestos utilizando Markdown:

Ejercicio 3

3.1. Crear en SQL todas las tablas.

```
CREATE DATABASE Video_Club_SQL;

USE Video_Club_SQL;

CREATE TABLE Socios (
    CodSocio INT AUTO_INCREMENT PRIMARY KEY,
    Nombre VARCHAR(25),
    Ciudad VARCHAR(15),
    Sexo BIT,
    Observaciones VARCHAR(50),
    Telefono CHAR(9)
);
```

```
CREATE TABLE Peliculas (
    CodPeli INT AUTO_INCREMENT PRIMARY KEY,
    Titulo VARCHAR(25),
    Tema VARCHAR(15),
    Nacionalidad VARCHAR(15),
    Fecha_Produccion DATE,
    Duracion INT,
    Precio DECIMAL(6, 2)
);

CREATE TABLE Alquiler (
    CodAlquiler INT AUTO_INCREMENT PRIMARY KEY,
    CodSocio INT,
    CodPeli INT,
    Fecha_Alquiler DATE,
    Fecha_Entrega DATE,
    Dias_Limite INT
);
```

3.2. Añadir una nueva columna de teléfono a la tabla de Socios.

```
ALTER TABLE Socios
ADD Telefono CHAR(9);
```

3.3. Modificar la tabla Alquiler para introducir claves primarias y claves ajenas.

```
ALTER TABLE Alquiler
DROP PRIMARY KEY,
ADD PRIMARY KEY (CodAlquiler),
ADD FOREIGN KEY (CodSocio) REFERENCES Socios(CodSocio),
ADD FOREIGN KEY (CodPeli) REFERENCES Peliculas(CodPeli);
```

3.4. Adjuntar la imagen correspondiente a las tablas y sus relaciones en el documento de texto.

Orden para introducir datos en las tablas:

1. Insertar datos en la tabla **Socios**.
2. Insertar datos en la tabla **Peliculas**.
3. Insertar datos en la tabla **Alquiler**.

Es importante insertar datos en las tablas relacionadas antes de insertar datos en las tablas que tienen claves ajenas a estas.

3.5. Obtener las siguientes consultas:

```
SELECT * FROM Alquiler;
```

```
SELECT Titulo, Nacionalidad  
FROM Peliculas  
WHERE Titulo LIKE '%a%' AND Nacionalidad LIKE '_R%';
```

```
SELECT CodPeli, Titulo, Precio  
FROM Peliculas  
WHERE Precio BETWEEN 10 AND 30  
ORDER BY Titulo DESC;
```