

Nombre:

Fecha:

Profesor:

Materia:

Institución:

Curso:

Nota:

Cliente

PK id_cliente
nombre
telefono

Taller

PK id-taller
nombre_taller
ubicacion

Servicio

id_servicio PK
descripcion
costo

Vehiculo

PK id_vehiculo
FK id_cliente
placa
marca

Vehiculo_Taller

PK FK id_vehiculo
PK FK id_taller
fecha_ingreso
fecha_salida

Reuestdo

PK id_reuestdo
nombre
precio

Servicio_Reuestdo

CK FK id_servicio
PK FK id_reuestdo
cantidad

CREATE DATABASE Taller;
USE Taller;

CREATE TABLE Cliente(
id_cliente INT PRIMARY KEY,
nombre VARCHAR(100),
telefono VARCHAR(20))
;

CREATE TABLE Vehiculo(
id_vehiculo INT PRIMARY KEY,
id_cliente INT,
placa VARCHAR(10),
marca VARCHAR(50),
FK (id_cliente) REFERENCES Cliente (id_cliente))
;

```

CREATE TABLE Taller(
    id_taller INT PRIMARY KEY,
    nombre_taller VARCHAR(100),
    direccion VARCHAR(200)
);

CREATE TABLE Vehiculo_Taller (
    id_vehiculo INT,
    id_taller INT,
    fecha_ingreso DATE,
    fecha_salida DATE,
    PRIMARY KEY (id_vehiculo, id_taller),
    FK(id_vehiculo) REFERENCES Vehiculo(id_vehiculo),
    FK(id_taller) REFERENCES Taller (id_taller)
);

CREATE TABLE Servicio(
    id_servicio INT PRIMARY KEY,
    descripcion VARCHAR(200),
    costo DECIMAL(10,2)
);

CREATE TABLE Repuesto(
    id_reuesto INT PRIMARY KEY,
    nombre VARCHAR(100),
    precio DECIMAL(10,2)
);

CREATE TABLE Servicio_Reuesto(
    id_servicio INT,
    id_reuesto INT,
    cantidad INT,
    PRIMARY KEY (id_servicio, id_reuesto),
    FOREIGN KEY (id_servicio) REFERENCES Servicio (id_servicio),
    FOREIGN KEY (id_reuesto) REFERENCES Repuesto (id_reuesto)
);

INSERT INTO Cliente VALUES
(1, 'Carlos Perez', '300123456789'),
(2, 'Laura Gomez', '301234567890');

INSERT INTO Vehiculo (id_vehiculo, id_cliente, placa, marca) VALUES
(10, 1, 'ABC123', 'Toyota'),
(20, 1, 'XYZ789', 'Mazda');

INSERT INTO Taller (id_taller, nombre_taller, ubicacion) VALUES
(101, 'Taller Motor', 'Cra 10 # 23-45'),
(102, 'AutoFix', 'Calle 50 # 15-30');

INSERT INTO Vehiculo_Taller (id_vehiculo, id_taller, fecha_ingreso, fecha_salida) VALUES
(10, 101, '2025-06-01', '2025-06-05');

INSERT INTO Servicio (id_servicio, descripcion, costo) VALUES
(201, 'Cambio de aceite', 80.00),
(202, 'Alineación y balanceo', 120.00);

```

```
INSERT INTO Repuesto (id-reuesto, nombre, precio) VALUES  
(301, 'Filtro de aceite', 35000),  
(302, 'Llanta r16 151, 180000);'  
  
INSERT INTO Servicio_Repuesto (id_servicio, id_reuesto, cantidad) VALUES  
(201, 301, 1);  
  
SELECT nombre  
FROM Cliente  
WHERE id-cliente IN  
    (SELECT id-cliente  
    FROM Vehiculo );  
  
SELECT placa  
FROM Vehiculo  
WHERE id-Vehiculo IN  
    (SELECT id-Vehiculo  
    FROM Vehiculo-Taller );  
  
SELECT v.placa, v.marca, c.nombre AS Cliente  
FROM Vehiculo v  
INNER JOIN Cliente c ON v.id-cliente = c.id-cliente;  
  
SELECT s.descripcion, r.nombre AS repuesto, sr.cantidad  
FROM Servicio s  
INNER JOIN Servicio_Repuesto sr ON s.id-Servicio = sr.id_servicio  
INNER JOIN Repuesto r ON sr.id-reuesto = r.id-reuesto;  
  
UPDATE Cliente SET telefono = '3029220261'  
WHERE id-cliente= 1;  
  
UPDATE Servicio SET costo = 95000  
WHERE id-Servicio= 201;  
  
DELETE FROM Repuesto WHERE id-reuesto= 302;  
DELETE FROM Vehiculo WHERE id-Vehiculo= 20;  
  
SELECT UPPER(nombre) AS nombre-mayuscula  
FROM Cliente;  
  
SELECT CONCAT(marca, ' ', modelo) AS vehiculo  
FROM Vehiculo;  
  
SELECT MAX(costo) AS costo_maximo_servicio  
FROM Servicio;  
  
ALTER TABLE Cliente ADD email VARCHAR(100);  
ALTER TABLE Taller DROP COLUMN ubicacion;  
  
TRUNCATE TABLE Servicio_Repuesto;  
TRUNCATE TABLE Vehiculo-Taller;  
  
DROP TABLE Cliente;  
DROP TABLE Vehiculo;
```

Socio

PK id_socio
nombre
apellido

Socio_Disciplina

PK FK id_socio
PK FK id-disciplina
fecha_inicio

Y

Reserva

PK id_reserva
FK id_sala
FK id_disciplina
fecha_reserva

Y

Sala

PK id_sala
nombre
capacidad

Disciplina

PK id_disciplina
nombre
categoria

Y

Monitor

PK FK id_monitor
PK FK id_disciplina
fecha_asignacion

Y

Monitor

PK id_monitor
nombre
especialidad

CREATE DATABASE Club;
USE Club;

CREATE TABLE Socio(
id_socio INT PRIMARY KEY,
nombre VARCHAR(50),
apellido VARCHAR(50),
);

CREATE TABLE Disciplina(
id_disciplina INT PRIMARY KEY,
nombre VARCHAR(50),
categoria VARCHAR(30)
);

CREATE TABLE Monitor(
id_monitor INT PRIMARY KEY,
nombre VARCHAR(50),
especialidad VARCHAR(50)
);

Nombre:

Fecha:

Profesor:

Materia:

Institución:

Curso:

Nota:

```
CREATE TABLE Sala (
    id_sala INT PRIMARY KEY,
    nombre VARCHAR(50),
    capacidad INT
);
```

```
CREATE TABLE Socio_Disciplina (
    id_socio INT,
    id_disciplina INT,
    fecha_inicio DATE,
    PRIMARY KEY (id_socio, id_disciplina),
    FK(id_socio) REFERENCES Socio (id_socio),
    FK(id_disciplina) REFERENCES Disciplina (id_disciplina)
);
```

```
CREATE TABLE Monitor_Disciplina (
    id_monitor INT,
    id_disciplina INT,
    fecha_asignacion DATE,
    PRIMARY KEY (id_monitor, id_disciplina),
    FOREIGN KEY (id_monitor) REFERENCES Monitor (id_monitor),
    FK(id_disciplina) REFERENCES Disciplina (id_disciplina)
);
```

```
CREATE TABLE Reserva (
    id_reserva INT PRIMARY KEY,
    id_sala INT,
    id_disciplina INT,
    fecha_reserva DATE,
    FK(id_sala) REFERENCES Sala (id_sala),
    FK(id_disciplina) REFERENCES Disciplina (id_disciplina)
);
```

```
INSERT INTO Socio VALUES
(1, 'Carlos', 'Gomez'),
(2, 'Lucia', 'Ramirez');
```

```
INSERT INTO Disciplina VALUES
(10, 'Natacion', 'Acuatica'),
(11, 'Tenis', 'Deporte de raqueta');
```

```
INSERT INTO Monitor VALUES
(100, 'Andres Torres', 'Natacion'),
(101, 'Diana Lopez', 'Tenis');
```

```
INSERT INTO Sala VALUES
(200, 'Piscina Olimpica', 30),
(201, 'Cancha de tenis', 4);
```

```
INSERT INTO Socio_Disciplina VALUES
(1, 10, '2025-06-01'),
(2, 11, '2025-06-02');
```

```
INSERT INTO Monitor_Disciplina VALUES  
(100, 10 '2025-05-15'),  
(101, 11 '2025-05-20');
```

```
INSERT INTO Reserva VALUES  
(1, 200, 10, '2025-07-10'),  
(2, 201, 11, '2025-07-11');
```

```
SELECT nombre  
FROM Socio  
WHERE id_socio IN  
(SELECT id_socio  
FROM Socio_Disciplina);
```

```
SELECT nombre  
FROM Disciplina  
WHERE id_disciplina IN  
(SELECT id_disciplina  
FROM Monitor_Disciplina);
```

```
SELECT nombre  
FROM Sala  
WHERE id_sala IN  
(SELECT id_sala  
FROM Reserva);
```

```
SELECT s.nombre, d.nombre AS disciplina  
FROM Socio s  
INNER JOIN Socio_Disciplina sd ON s.id_socio = sd.id_socio  
INNER JOIN Disciplina d ON sd.id_disciplina = d.id_disciplina;
```

```
SELECT r.fecha_reserva, sa.nombre AS sala, d.nombre  
FROM Reserva r  
INNER JOIN Sala sa ON r.id_sala = sa.id_sala  
INNER JOIN Disciplina d ON r.id_disciplina = d.id_disciplina;
```

```
UPDATE Socio SET apellido = 'Martinez'  
WHERE id_socio = 1;
```

```
UPDATE Sala SET capacidad = 50  
WHERE id_sala = 200;
```

```
DELETE FROM Reserva WHERE id_reserva = 1;  
DELETE FROM Monitor WHERE id_monitor = 101;
```

```
SELECT CONCAT(nombre, ' ', apellido) AS nombre_completo  
FROM Socio
```

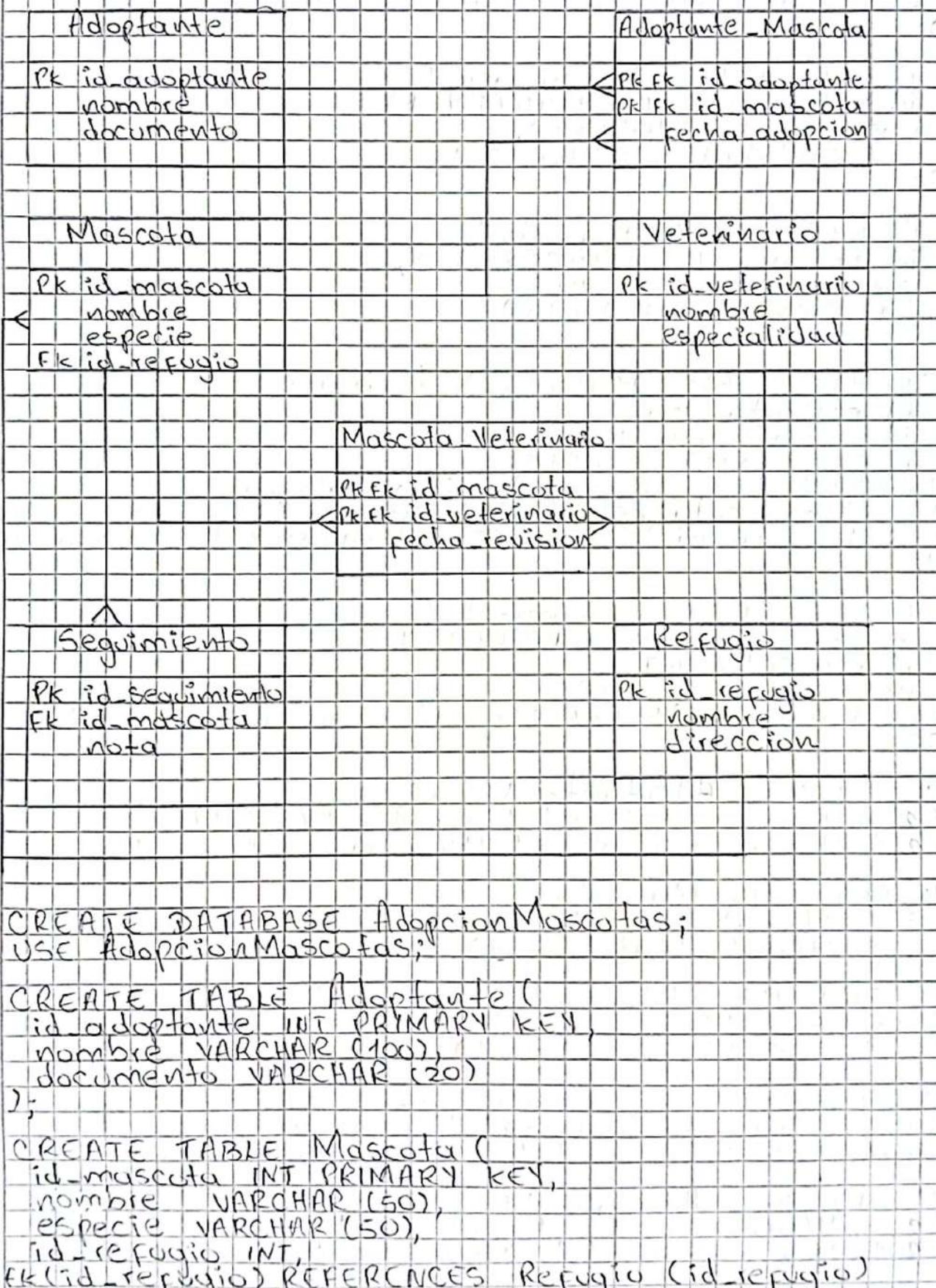
```
SELECT MIN(capacidad) AS capacidad_minima  
FROM Sala;
```

```
SELECT UPPER(nombre) AS nombre_mayuscula  
FROM Monitor;
```

```
ALTER TABLE Socio ADD email VARCHAR(50);  
ALTER TABLE Reserva DROP COLUMN fecha_reserva;
```

TRUNCATE TABLE Monitor_Disciplina;
TRUNCATE TABLE Reserva;

DROP TABLE Sala;
DROP TABLE Socio-Disciplina;



```
CREATE TABLE Adoptante_Mascota (
    id_adoptante INT,
    id_mascota INT,
    fecha_adopcion DATE,
PRIMARY KEY (id_adoptante, id_mascota),
FK (id_adoptante) REFERENCES Adoptante (id_adoptante),
FK (id_mascota) REFERENCES Mascota (id_mascota)
);
```

```
CREATE TABLE Veterinario (
    id_veterinario INT PRIMARY KEY,
    nombre VARCHAR(100),
    especialidad VARCHAR(50)
);
```

```
CREATE TABLE Refugio (
    id_refugio INT PRIMARY KEY,
    nombre VARCHAR(100),
    direccion VARCHAR(100)
);
```

```
CREATE TABLE Mascota_Veterinario (
    id_mascota INT,
    id_veterinario INT,
    fecha_revision DATE,
PRIMARY KEY (id_mascota, id_veterinario),
FK (id_mascota) REFERENCES Mascota (id_mascota),
FK (id_veterinario) REFERENCES Veterinario (id_veterinario)
);
```

```
CREATE TABLE Seguimiento (
    id_seguimiento INT PRIMARY KEY,
    id_mascota INT,
    nota VARCHAR(100),
FK (id_mascota) REFERENCES Mascota (id_mascota)
);
```

```
INSERT INTO Refugio VALUES
(100, 'Refugio Pafititas', 'Calle 123 #45-671'),
(101, 'Hoyate Animalito', 'Avenida 10 #20-30');
```

```
INSERT INTO Mascota VALUES
(300, 'Luna', 'Perro', 100),
(301, 'Milo', 'Gato', 101);
```

```
INSERT INTO Adoptante_Mascota VALUES
(200, 300, '2025-07-01'),
(201, 301, '2025-07-05');
```

```
INSERT INTO Veterinario VALUES
(400, 'Dra. Camila Perez', 'Medicina Felina'),
(401, 'Dr. Andres Soto', 'Medicina Canina');
```

```
INSERT INTO Adoptante VALUES
(100, 'Maria Lopez', '1234567891'),
(201, 'Carlos Ramirez', '987654321');
```

Nombre:

Fecha:

Profesor:

Materia:

Institución:

Curso:

Nota:

INSERT INTO Mascota_Veterinario (id_mascota, id_veterinario, fecha_revision) VALUES

(300, 401, '2025-06-20'),

(301, 400, '2025-06-22');

INSERT INTO Seguimiento VALUES

(500, 300, 'Mascota se encuentra bien'),

(501, 301, 'Requiere vacunación el otro mes');

SELECT nombre

FROM Adoptante

WHERE id_adoptante IN

SELECT id_adoptante

FROM Adoptante_Mascota);

SELECT nombre

FROM Refugio

WHERE id_refugio IN

SELECT id_refugio

FROM Mascota);

SELECT m.nombre AS Mascota, v.nombre AS veterinario

FROM Mascota m

INNER JOIN Mascota_Veterinario mv ON m.id_mascota =

mv.id_mascota

INNER JOIN Veterinario v ON mv.id_veterinario =

v.id_veterinario;

SELECT m.nombre AS mascota, r.nombre AS refugio

FROM mascota m

INNER JOIN Refugio r ON m.id_refugio = r.id_refugio

UPDATE Refugio SET dirección = 'Calle Nueva # 10-20'

WHERE id_refugio = 100;

UPDATE Veterinario SET especialidad = 'Cirugía'

WHERE id_veterinario = 401;

DELETE FROM Seguimiento WHERE id_seguimiento = 501;

DELETE FROM Mascota WHERE id_mascota = 301;

SELECT UPPER(nombre) AS nombre_mayuscula

FROM Mascota;

SELECT CONCAT(nombre, ' ', documento) AS información

FROM Adoptante;

ALTER TABLE Adoptante ADD telefono VARCHAR(10);

ALTER TABLE Mascota DROP COLUMN especie;

TRUNCATE TABLE Seguimiento;
TRUNCATE TABLE Mascota_Veterinario;

DROP TABLE Adoptante_Mascota;
DROP TABLE Refugio;

Cliente
PK id_cliente
nombre
correo

Factura
PK id_factura
FK id_cliente
FK id_empleado
fecha

Empleado
PK id_empleado
nombre
Cargo

Producto
PK id_producto
nombre
precio

Factura_Producto
PK FK id_factura
PK FK id_producto
cantidad

Proveedor
PK id_proveedor
nombre
contacto

Producto_Proveedor
PK FK id_producto
PK FK id_proveedor

CREATE DATABASE TiendaRopa;
USE TiendaRopa;

CREATE TABLE Cliente (
id_cliente INT PRIMARY KEY,
nombre VARCHAR(100),
correo VARCHAR(100)
);

CREATE TABLE Producto (
id_producto INT PRIMARY KEY,
nombre VARCHAR(100),
precio DECIMAL(10,2)
);

CREATE TABLE Empleado (
id_empleado INT PRIMARY KEY,
nombre VARCHAR(100),
cargo VARCHAR(50)
);

CREATE TABLE Factura (
id_factura INT PRIMARY KEY,
id_cliente INT,
id_empleado INT,
fecha DATE,

FK(id_cliente) REFERENCES Cliente(id_cliente),
FK(id_empleado) REFERENCES Empleado(id_empleado)
);

CREATE TABLE Proveedor (
id_proveedor INT PRIMARY KEY,
nombre VARCHAR(100),
contacto VARCHAR(100)
);

CREATE TABLE Factura_Producto (
id_factura INT,
id_producto INT,
cantidad INT,
PRIMARY KEY (id_factura, id_producto),

CREATE TABLE Producto_Proveedor (
id_producto INT,
id_proveedor INT,
PRIMARY KEY (id_producto, id_proveedor),
FK(id_producto) REFERENCES Producto(id_producto),
FK(id_proveedor) REFERENCES Proveedor(id_proveedor)
);

INSERT INTO Cliente VALUES
(100, 'Laura Torres', 'laura@mail.com'),
(101, 'Juan Martinez', 'juan@mail.com');

INSERT INTO Producto VALUES
(200, 'Camisa Blanca', 45000.00),
(201, 'Pantalon Negro', 80000.00);

INSERT INTO Empleado VALUES
(300, 'Sofia Herrera', 'Vendedor'),
(301, 'Carlos Diaz', 'Cajero'));

INSERT INTO Factura VALUES
(400, 100, 300, '2025-07-01'),
(401, 101, 301, '2025-07-02');

INSERT INTO Proveedor VALUES
(500, 'Textiles S.A.S.', 'textiles@proveedor.com'),
(501, 'RopaFina Ltda.', 'ropafina@proveedor.com');

INSERT INTO Factura_Producto VALUES
(400, 200, 2),
(401, 201, 1);

INSERT INTO Producto_Proveedor VALUES
(200, 500),
(201, 501);

```
SELECT nombre  
FROM Producto  
WHERE id_producto IN  
    SELECT id_producto  
    FROM Factura_Producto );
```

```
SELECT nombre  
FROM Proveedor  
WHERE id_proveedor IN  
    SELECT id_proveedor  
    FROM Producto_Proveedor  
    WHERE id_producto IN  
        SELECT id_producto  
        FROM Producto  
        WHERE precio = 45000.00 ) );
```

```
SELECT f.id_factura, c.nombre AS cliente, e.nombre AS empleado  
FROM Factura f  
INNER JOIN Cliente c ON f.id_cliente = c.id_cliente  
INNER JOIN Empleado e ON f.id_empleado = e.id_empleado;
```

```
SELECT p.nombre AS producto, pr.nombre AS proveedor  
FROM Producto_Proveedor pp  
INNER JOIN Producto p ON pp.id_producto = p.id_producto  
INNER JOIN Proveedor pr ON pp.id_proveedor = pr.id_proveedor;
```

```
UPDATE Producto SET precio = 49000.00  
WHERE id_producto = 200;
```

```
UPDATE Empleado SET cargo = 'Gerente'  
WHERE id_empleado = 301;
```

```
DELETE FROM Cliente WHERE id_cliente = 101;  
DELETE FROM Factura WHERE id_factura = 400;
```

```
SELECT MAX(precio) AS precio_maximo FROM Producto;
```

```
SELECT CONCAT(nombre, ' - ' cargo) AS info FROM Empleado;
```

```
ALTER TABLE Cliente ADD telefono VARCHAR(10);  
ALTER TABLE Proveedor DROP COLUMN contacto;
```

```
TRUNCATE TABLE Factura_Proveedor;
```

```
TRUNCATE TABLE Empleado;
```

```
DROP TABLE Cliente;
```

```
DROP TABLE Producto;
```

Nombre:

Fecha:

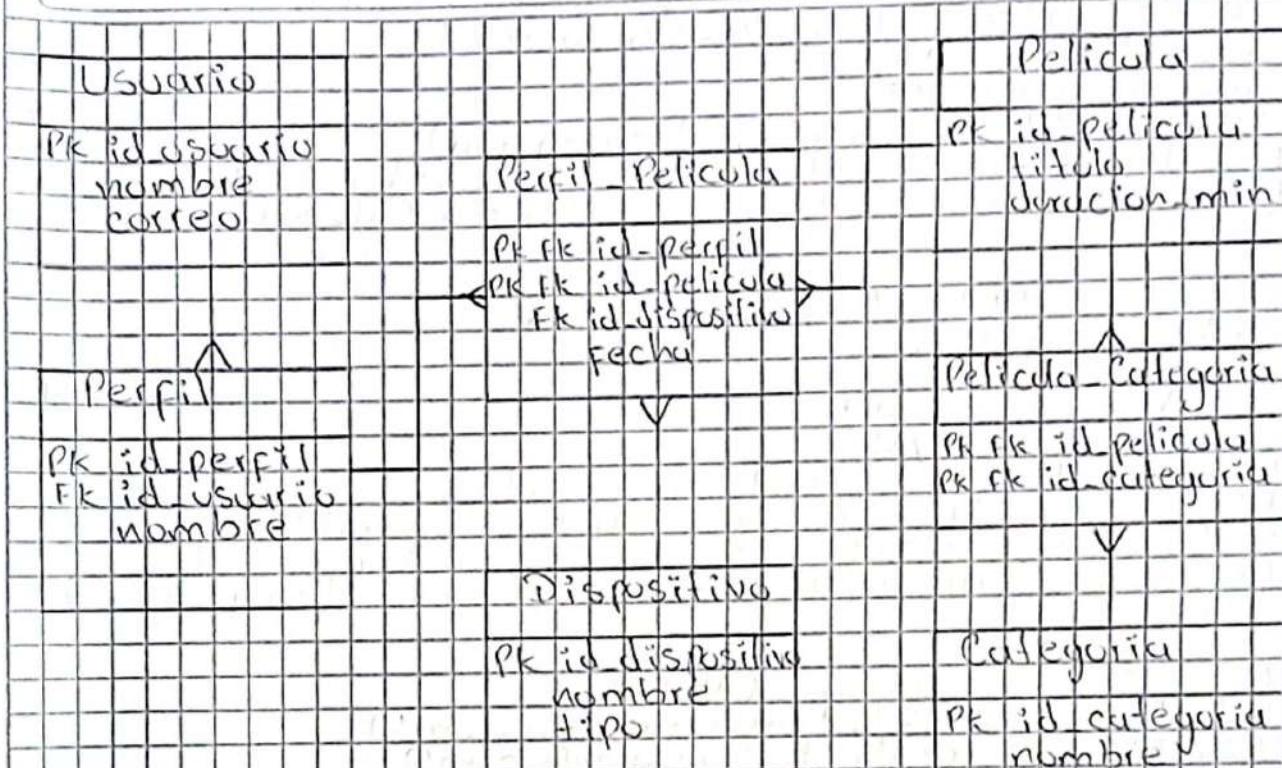
Profesor:

Materia:

Institución:

Curso:

Nota:



```

CREATE DATABASE PlataformaStreaming;
USE PlataformaStreaming;

```

```

CREATE TABLE Usuario (
    id_usuario INT PRIMARY KEY,
    nombre VARCHAR(100),
    correo VARCHAR(100)
);

```

```

CREATE TABLE Perfil (
    id_perfil INT PRIMARY KEY,
    id_usuario INT,
    nombre VARCHAR(100),
    FOREIGN KEY(id_usuario) REFERENCES Usuario(id_usuario)
);

```

```

CREATE TABLE Pelicula (
    id_pelicula INT PRIMARY KEY,
    titulo VARCHAR(100),
    duracion_min INT
);

```

```

CREATE TABLE Categoria (
    id_categoria INT PRIMARY KEY,
    nombre VARCHAR(50)
);

```

```
CREATE TABLE Dispositivo (
    id_dispositivo INT PRIMARY KEY,
    nombre VARCHAR(100),
    tipo VARCHAR(50)
);
```

```
CREATE TABLE Pelicula_Categoría (
    id_pelicula INT,
    id_categoria INT,
    PRIMARY KEY (id_pelicula, id_categoria),
    FK(id_pelicula) REFERENCES Pelicula(id_pelicula),
    FK(id_categoria) REFERENCES Categoría(id_categoria)
);
```

```
CREATE TABLE Perfil_Pelicula (
    id_perfil INT,
    id_pelicula INT,
    id_dispositivo INT,
    fecha DATETIME,
    PRIMARY KEY (id_perfil, id_pelicula),
    FK(id_perfil) REFERENCES Perfil(id_perfil),
    FK(id_pelicula) REFERENCES Pelicula(id_pelicula),
    FK(id_dispositivo) REFERENCES Dispositivo(id_dispositivo)
);
```

```
INSERT INTO Usuario VALUES
(1, 'Ana Gomez', 'ana@example.com'),
(2, 'Luis Torres', 'luis@example.com');
```

```
INSERT INTO Perfil VALUES
(10, 1, 'Ana Kids'),
(11, 2, 'Luis HD');
```

```
INSERT INTO Pelicula VALUES
(100, 'Inception', 148),
(101, 'Toy Story', 86);
```

```
INSERT INTO Categoría VALUES
(200, 'Ciencia ficción'),
(201, 'Animación');
```

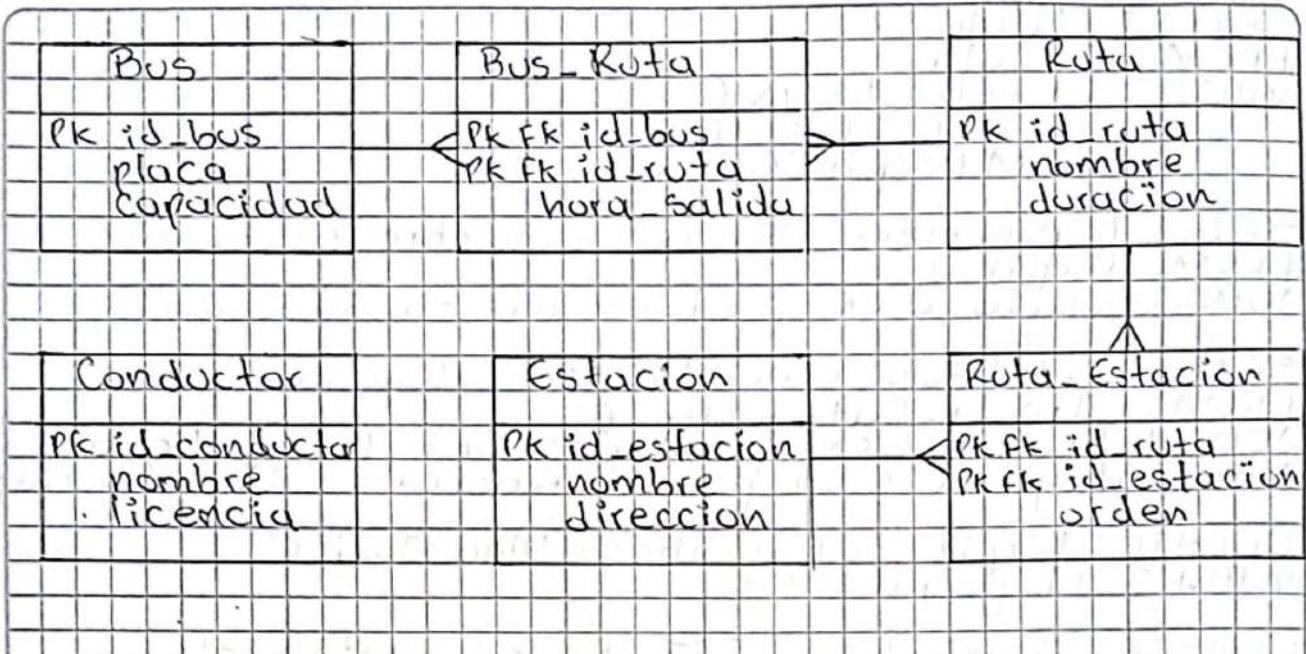
```
INSERT INTO Dispositivo VALUES
(300, 'Smart TV Samsung', 'TV'),
(301, 'Phone 14', 'Móvil');
```

```
INSERT INTO Pelicula_Categoría VALUES
(100, 200),
(101, 201);
```

```
INSERT INTO Perfil_Pelicula VALUES
(10, 101, 301, '2025-07-08 18:30:00'),
(11, 100, 300, '2025-07-07 21:00:00');
```

```
SELECT nombre
FROM Usuario
WHERE id_usuario IN (
    SELECT id_usuario
    FROM Perfil
);
```

```
SELECT titulo
FROM Pelicula
WHERE id_pelicula IN (
    SELECT id_pelicula
    FROM Perfil_Pelicula );
SELECT p.nombre AS perfil, u.nombre AS usuario
FROM Perfil p
JOIN Usuario u ON p.id_usuario = u.id_usuario;
SELECT pl.titulo, c.nombre AS categoria
FROM Pelicula_Categoría pc
JOIN Pelicula pl ON pc.id_pelicula = pl.id_pelicula
JOIN Categoría c ON pc.id_categoria = c.id_categoria;
UPDATE Perfil SET nombre = 'Ana Junior'
WHERE id_perfil = 10;
UPDATE Dispositivo SET tipo = 'Smart phone'
WHERE id_dispositivo = 301;
DELETE FROM Categoría WHERE id_categoria = 201;
DELETE FROM Dispositivo WHERE id_dispositivo = 300;
SELECT CONCAT(nombre, ' - ', correo) AS usuario_info
FROM Usuario;
SELECT AVG(duracion_min) AS promedio_duracion
FROM Pelicula;
ALTER TABLE Usuario ADD pais VARCHAR(20);
ALTER TABLE Dispositivo DROP COLUMN tipo;
TRUNCATE TABLE Perfil_Pelicula;
TRUNCATE TABLE Dispositivo;
DROP TABLE Categoría;
DROP TABLE Usuario;
```



Pasajero

PK id_pasajero
nombre
tarjeta_id

CREATE DATABASE Transporte;
USE Transporte;

CREATE TABLE Pasajero (
id_pasajero INT PRIMARY KEY,
nombre VARCHAR(100),
tarjeta_id VARCHAR(50))
;

CREATE TABLE Ruta (
id_ruta INT PRIMARY KEY,
nombre VARCHAR(100),
duración INT);

CREATE TABLE Bus (
id_bus INT PRIMARY KEY,
placa VARCHAR(10),
capacidad INT)
;

CREATE TABLE Conductor (
id_conductor INT PRIMARY KEY,
nombre VARCHAR(100),
licencia VARCHAR(50))
;

CREATE TABLE Estacion (
id_estacion INT PRIMARY KEY,
nombre VARCHAR(100),
direccion VARCHAR(100))
;

Nombre:

Fecha:

Profesor:

Materia:

Institución:

Curso:

Nota:

```
CREATE TABLE Bus_Ruta (
    id_bus INT,
    id_ruta INT,
    hora_salida TIME,
    PRIMARY KEY (id_bus, id_ruta),
    FK(id_bus) REFERENCES Bus (id_bus),
    FK(id_ruta) REFERENCES Ruta (id_ruta)
);
```

```
CREATE TABLE Ruta_Estacion (
    id_ruta INT,
    id_estacion INT,
    orden INT,
    PRIMARY KEY (id_ruta, id_estacion),
    FK(id_ruta) REFERENCES Ruta (id_ruta),
    FK(id_estacion) REFERENCES Estacion (id_estacion)
);
```

```
INSERT INTO Pasajero VALUES
(1, 'Maria Rodriguez', 'TAR123'),
(2, 'Javier Mendez', 'TAR456');
```

```
INSERT INTO Ruta VALUES
(10, 'Ruta Norte', 45),
(11, 'Ruta Sur', 60);
```

```
INSERT INTO Bus VALUES
(100, 'ABC-123', 40),
(101, 'XYZ-789', 50);
```

```
INSERT INTO Conductor VALUES
(200, 'Luis Torres', 'LIC987'),
(201, 'Diana Gomez', 'LIC887');
```

```
INSERT INTO Estacion VALUES
(300, 'Estacion Central', 'Cra 10 #20-30'),
(301, 'Parada Norte', 'Av 15 #45-60');
```

```
INSERT INTO Bus_Ruta VALUES
(100, 10, '06:30:00'),
(101, 11, '07:15:00');
```

```
INSERT INTO Ruta_Estacion VALUES
(10, 300, 1),
(11, 301, 2);
```

```
SELECT nombre
FROM Ruta
WHERE id_ruta IN (
    SELECT id_ruta
    FROM Bus_Ruta
);
```

```
SELECT nombre
FROM Estacion
WHERE id_estacion IN (
    SELECT id_estacion
    FROM Ruta_Estacion
    WHERE id_ruta IN (
        SELECT id_ruta
        FROM Ruta
        WHERE nombre = 'Ruta Sur'
    )
);

SELECT b.placa, r.nombre AS ruta
FROM Bus b
INNER JOIN Bus_Ruta br ON b.id_bus = br.id_bus
INNER JOIN Ruta r ON br.id_ruta = r.id_ruta;

SELECT c.nombre AS conductor, b.placa
FROM Conductor c
INNER JOIN Bus b ON c.id_conductor = b.id_bus;

UPDATE Bus SET capacidad = 50 WHERE id_bus = 100;

UPDATE Estacion SET nombre = 'Estacion Norte'
WHERE id_estacion = 301;

DELETE FROM Conductor WHERE id_conductor = 200;

DELETE FROM Estacion WHERE id_estacion = 300;

SELECT MAX(duracion_estimada) AS duracion_maxima
FROM Ruta;

SELECT MIN(capacidad) AS capacidad_menor
FROM Bus;

ALTER TABLE Pasajero ADD edad INT;

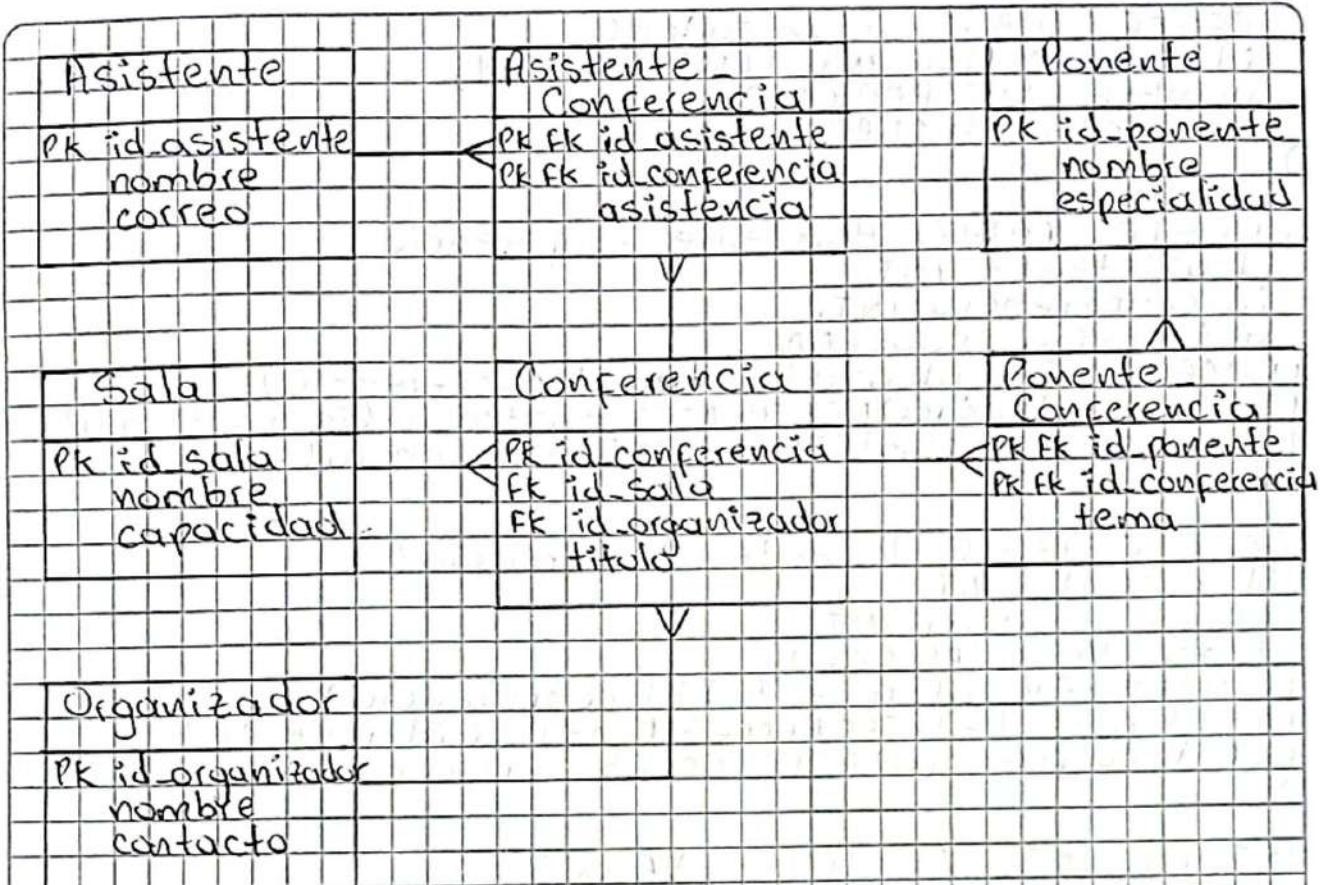
ALTER TABLE Conductor DROP COLUMN licencia;

TRUNCATE TABLE Bus_Ruta;

TRUNCATE TABLE Conductor;

DROP TABLE Pasajero;

DROP TABLE Ruta_Estacion;
```



CREATE DATABASE Eventos;
 USE Eventos;

```
CREATE TABLE Asistente (
  id_asistente INT PRIMARY KEY,
  nombre VARCHAR(100),
  correo VARCHAR(100)
);
```

```
CREATE TABLE Conferencia (
  id_conferencia INT PRIMARY KEY,
  id_sala INT,
  id_organizador INT,
  titulo VARCHAR(50),
  FK(id_sala) REFERENCES Sala (id_sala),
  FK(id_organizador) REFERENCES Organizador (id_organizador)
);
```

```
CREATE TABLE Ponente (
  id_ponente INT PRIMARY KEY,
  nombre VARCHAR(100),
  especialidad VARCHAR(100)
);
```

```
CREATE TABLE Sala (
  id_sala INT PRIMARY KEY,
  nombre VARCHAR(50),
  capacidad INT
);
```

```
CREATE TABLE Organizador (
    id_organizador INT PRIMARY KEY,
    nombre VARCHAR(100),
    contacto VARCHAR(50)
);
CREATE TABLE Asistente_Conferencia (
    id_asistente INT,
    id_conferencia INT,
    asistencia BOOLEAN
PRIMARY KEY (id_asistente, id_conferencia),
FK(id_asistente) REFERENCES Asistente(id_asistente),
FK(id_conferencia) REFERENCES Conferencia(id_conferencia)
);
```

```
CREATE TABLE Ponente_Conferencia (
    id_ponente INT,
    id_conferencia INT,
    tema VARCHAR(100)
PRIMARY KEY (id_ponente, id_conferencia),
FK(id_ponente) REFERENCES Ponente(id_ponente),
FK(id_conferencia) REFERENCES Conferencia(id_conferencia)
);
```

```
INSERT INTO Asistente VALUES
(1, 'Laura Pérez', 'laura@example.com'),
(2, 'Carlos Ríos', 'carlos@example.com');
```

```
INSERT INTO Sala VALUES
(10, 'Auditoria A', 150),
(11, 'Sala B', 80);
```

```
INSERT INTO Organizador VALUES
(100, 'Mariana López', 'mariana@eventos.com'),
(101, 'Sergio Díaz', 'sergio@conferencias.com');
```

```
INSERT INTO Conferencia VALUES
(1000, 10, 100, 'Transformación Digital'),
(1001, 11, 101, 'Ciberseguridad');
```

```
INSERT INTO Ponente VALUES
(200, 'Dr. Andrés Vega', 'Transformación Digital'),
(201, 'Ing. Paula León', 'Ciberseguridad');
```

```
INSERT INTO Asistente_Conferencia VALUES
(1, 1000, TRUE),
(2, 1001, FALSE);
```

```
INSERT INTO Ponente_Conferencia VALUES
(200, 1000, 'Innovación en la era digital'),
(201, 1001, 'Seguridad zero Trust');
```

```
SELECT titulo
FROM Conferencia
WHERE id_conferencia IN (
    SELECT id_conferencia
    FROM Ponente_Conferencia );
```

Nombre:

Fecha:

Profesor:

Materia:

Institución:

Curso:

Nota:

```
SELECT nombre
FROM Sala
WHERE id_Sala IN (
    SELECT id_Sala
    FROM Conferencia
    WHERE id_Conferencia IN (
        SELECT id_Conferencia
        FROM Asistente_Conferencia
        WHERE asistencia = FALSE
    )
);

SELECT c.titulo, o.nombre AS organizador
FROM Conferencia c
INNER JOIN Organizador o ON c.id_organizador = o.id_organizador;

SELECT p.nombre AS ponente, cf.titulo, pc.tema
FROM Ponente_Conferencia pc
INNER JOIN Ponente p ON pc.id_ponente = p.id_ponente
INNER JOIN Conferencia cf ON pc.id_Conferencia = cf.id_Conferencia;

UPDATE Conferencia SET titulo = 'Tecnologías Emergentes'
WHERE id_Conferencia = 1000;

UPDATE Organizador SET contacto = 'mariangela.club.com'
WHERE id_Organizador = 100;

DELETE FROM Asistente WHERE id_asistente = 2;

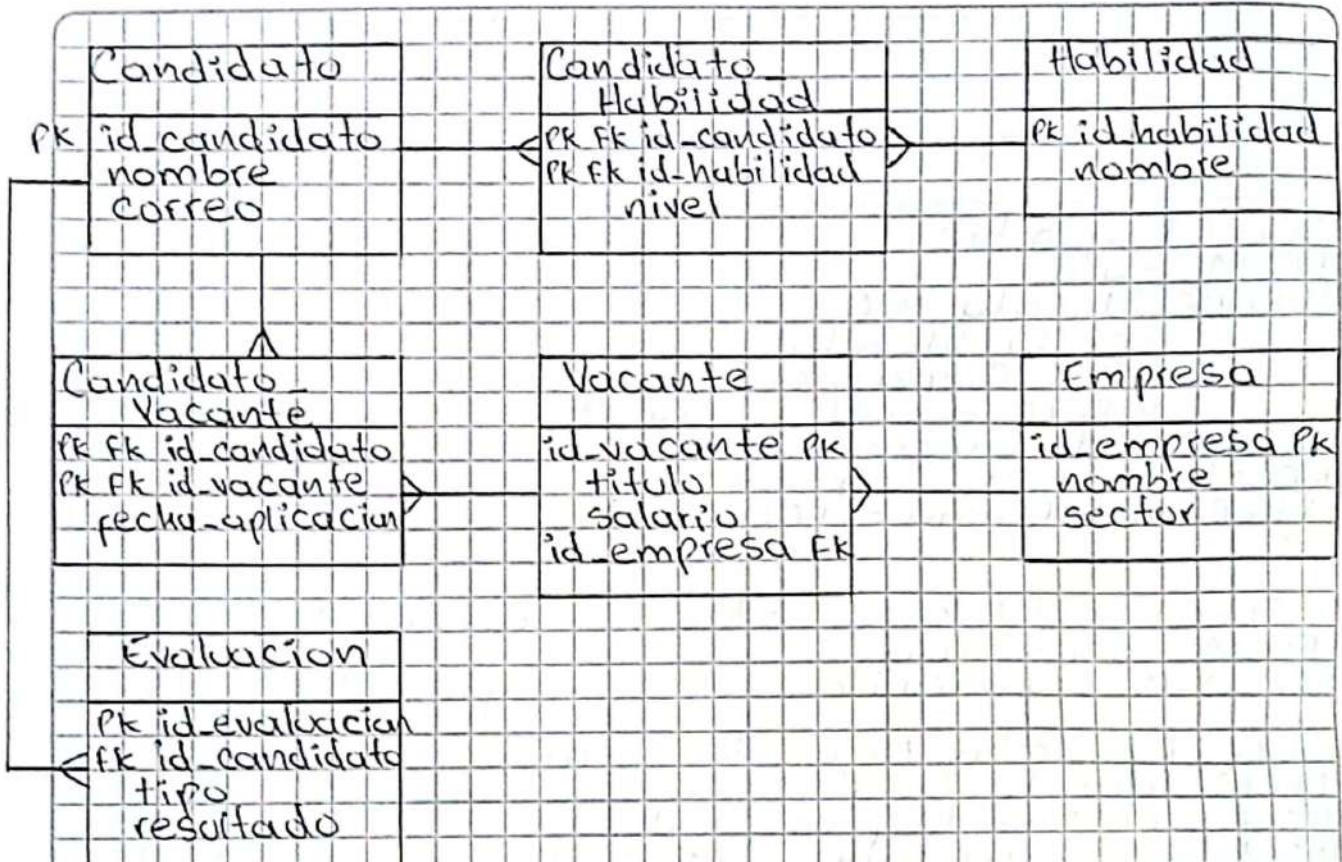
DELETE FROM Sala WHERE id_Sala = 11;

SELECT UPPER(nombre) AS ponente_mayuscula
FROM Ponente;

SELECT CONCAT(nombre, '- ', correo) AS info
FROM Asistente;

ALTER TABLE Organizador DROP COLUMN contacto;
ALTER TABLE Asistente ADD telefono VARCHAR(10);

TRUNCATE TABLE Ponente_Conferencia;
TRUNCATE TABLE Sala;
DROP TABLE Organizador;
DROP TABLE Asistente;
```



CREATE DATABASE Reclutamiento;

USE Reclutamiento;

```
CREATE TABLE Candidato (
    id_candidato INT PRIMARY KEY,
    nombre VARCHAR(100),
    correo VARCHAR(100)
);
```

```
CREATE TABLE Empresa (
    id_empresa INT PRIMARY KEY,
    nombre VARCHAR(100),
    sector VARCHAR(50)
);
```

```
CREATE TABLE Vacante (
    id_vacante INT PRIMARY KEY,
    titulo VARCHAR(100),
    salario DECIMAL(10,2),
    id_empresa INT,
    FK(id_empresa) REFERENCES Empresa (id_empresa)
);
```

```
CREATE TABLE Habilidad (
    id_habilidad INT PRIMARY KEY,
    nombre VARCHAR(50)
);
```

```
CREATE TABLE Evaluacion (
    id_evaluacion INT PRIMARY KEY,
    id_candidato INT,
    tipo VARCHAR(50),
    resultado VARCHAR(50),
    fk(id_candidato) REFERENCES Candidato(id_candidato)
);
```

```
CREATE TABLE Candidato_Habilidad (
    id_candidato INT,
    id_habilidad INT,
    nivel VARCHAR(20),
    PRIMARY KEY (id_candidato, id_habilidad),
    FK(id_candidato) REFERENCES Candidato(id_candidato),
    FK(id_habilidad) REFERENCES Habilidad(id_habilidad)
);
```

```
CREATE TABLE Candidato_Vacante (
    id_candidato INT,
    id_vacante INT,
    fecha_aplicacion DATE,
    PRIMARY KEY (id_candidato, id_vacante),
    FK(id_candidato) REFERENCES Candidato(id_candidato),
    FK(id_vacante) REFERENCES Vacante(id_vacante)
);
```

```
INSERT INTO Candidato VALUES
(1, 'Lucia Ramirez', 'lucia@gmail.com'),
(2, 'Andres Torres', 'andres@gmail.com');
```

```
INSERT INTO Empresa VALUES
(10, 'TechNova', 'Tecnología'),
(11, 'Financecorp', 'Finanzas');
```

```
INSERT INTO Vacante VALUES
(100, 'Desarrollador Backend', 4500000, 10),
(101, 'Analista Financiero', 3800000, 11);
```

```
INSERT INTO Habilidad VALUES
(200, 'Java'),
(201, 'Excel Avanzado');
```

```
INSERT INTO Evaluacion VALUES
(300, 1, 'Entrevista Técnica', 'Aprobado'),
(301, 2, 'Prueba Psicotécnica', 'Pendiente');
```

```
INSERT INTO Candidato_Habilidad VALUES
(1, 200, 'Avanzado'),
(2, 201, 'Intermedio');
```

```
INSERT INTO Candidato_Vacante VALUES
(1, 100, '2025-07-01'),
(2, 101, '2025-07-03');
```

```
SELECT nombre
FROM Candidato
WHERE id_candidato IN
    (SELECT id_candidato FROM Candidato_Vacante);
```

```
SELECT titulo  
FROM Vacante  
WHERE id_empresa IN(  
    SELECT id_empresa  
    FROM Empresa  
    WHERE nombre = 'TechNova' );  
  
SELECT c.nombre, h.nombre AS habilidad, ch.nivel  
FROM Candidato c  
INNER JOIN Candidato_Habilidad ch ON c.id_candidato = ch.id_candidato  
INNER JOIN Habilidad h ON ch.id_habilidad = h.id_habilidad;  
  
SELECT v.titulo, e.nombre AS empresa  
FROM Vacante v  
INNER JOIN Empresa e ON v.id_empresa = e.id_empresa  
  
UPDATE Vacante SET salario = 5000000  
WHERE id_vacante = 100;  
  
UPDATE Evaluacion SET resultado = 'Aprobado'  
WHERE id_evaluacion = 301;  
  
DELETE FROM Empresa WHERE id_empresa = 11;  
  
DELETE FROM Vacante WHERE id_vacante = 101;  
  
SELECT UPPER(nombre) AS nombre_mayuscula  
FROM Candidato;  
  
SELECT CONCAT(nombre, ' ', correo) AS datos  
FROM Candidato;  
  
ALTER TABLE Empresa ADD telefono VARCHAR(20);  
ALTER TABLE Evaluacion DROP COLUMN resultado;  
TRUNCATE TABLE Candidato_Vacante;  
TRUNCATE TABLE Evaluacion;  
DROP TABLE Candidato_Habilidad;  
DROP TABLE Vacante;
```

Nombre:

Fecha:

Profesor:

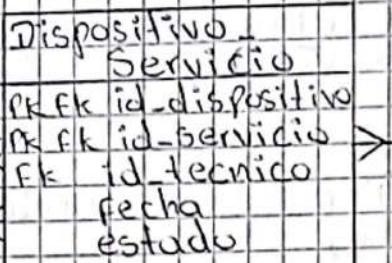
Materia:

Institución:

Curso:

Nota:

Dispositivo		
PK	id_dispositivo	
tipo		
marca		
fk_id_cliente		



Servicio		
PK	id_servicio	
	descripcion	
	costo	

Cliente		
PK	id_cliente	
	nombre	
	telefono	

Reuesto		
PK	id_reuesto	
	nombre	
	precio	

Servicio_Reuesto		
PK	id_servicio	
	id_reuesto	
	cantidad	

Tecnico

PK	id_tecnico	
	nombre	
	especialidad	

```
CREATE DATABASE ReparacionElectronica;
USE ReparacionElectronica;
```

```
CREATE TABLE Cliente (
    id_cliente INT PRIMARY KEY,
    nombre VARCHAR(100),
    telefono VARCHAR(20)
);
```

```
CREATE TABLE Dispositivo (
    id_dispositivo INT PRIMARY KEY,
    tipo VARCHAR(50),
    marca VARCHAR(50),
    fk_id_cliente INT,
    FK fk_id_cliente REFERENCES Cliente(id_cliente)
);
```

```
CREATE TABLE Tecnico (
    id_tecnico INT PRIMARY KEY,
    nombre VARCHAR(100),
    especialidad VARCHAR(50)
);
```

```

CREATE TABLE Servicio (
    id_servicio INT PRIMARY KEY,
    descripcion VARCHAR(100),
    costo DECIMAL(10,2)
);

CREATE TABLE Repuesto (
    id_reuesto INT PRIMARY KEY,
    nombre VARCHAR(50),
    precio DECIMAL(10,2)
);

CREATE TABLE Dispositivo_Servicio (
    id_dispositivo INT,
    id_servicio INT,
    id_tecnico INT,
    fecha DATE,
    estado VARCHAR(20),
    PRIMARY KEY (id_dispositivo, id_servicio),
    FK(id_dispositivo) REFERENCES Dispositivo(id_dispositivo),
    FK(id_servicio) REFERENCES Servicio(id_servicio),
    FK(id_tecnico) REFERENCES Tecnico(id_tecnico)
);

CREATE TABLE Servicio_Repuesto (
    id_servicio INT,
    id_reuesto INT,
    cantidad INT,
    PRIMARY KEY (id_servicio, id_reuesto),
    FK(id_servicio) REFERENCES Servicio(id_servicio),
    FK(id_reuesto) REFERENCES Repuesto(id_reuesto)
);

INSERT INTO Cliente VALUES
(1, 'Carlos Lopez', '3101234567'),
(2, 'Marta Gomez', '3209876543');

INSERT INTO Dispositivo VALUES
(100, 'Celular', 'Samsung', 1),
(101, 'Laptop', 'HP', 2);

INSERT INTO Tecnico VALUES
(200, 'Luis Ramirez', 'Hardware'),
(201, 'Ana Torres', 'Software');

INSERT INTO Servicio VALUES
(300, 'Cambio de pantalla', 180000),
(301, 'Formato y optimización', 90000);

INSERT INTO Repuesto VALUES
(400, 'Pantalla Samsung A12', 120000),
(401, 'SSD 256 GB', 150000);

INSERT INTO Dispositivo_Servicio VALUES
(100, 300, 200, '2025-07-01', 'Completo'),
(101, 301, 201, '2025-07-02', 'En proceso');

```

```
INSERT INTO Servicio_Repuesto VALUES  
(300, 400, 1),  
(301, 401, 1);  
  
SELECT nombre  
FROM Cliente  
WHERE id_cliente IN  
    (SELECT id_cliente  
     FROM Dispositivo );  
  
SELECT descripcion FROM Servicio  
WHERE id_servicio IN  
    (SELECT id_servicio FROM Dispositivo_Servicio  
     WHERE id_dispositivo IN  
        (SELECT id_dispositivo FROM Dispositivo  
         WHERE marca = 'HP' )  
    );  
  
SELECT c.nombre, d.tipo, d.marca  
FROM Cliente c  
JOIN Dispositivo d ON c.id_cliente = d.id_cliente;  
  
SELECT s.descripcion, r.nombre AS repuesto, sr.cantidad  
FROM Servicio_Repuesto sr  
JOIN Servicio s ON sr.id_servicio = s.id_servicio  
JOIN Repuesto r ON sr.id_repuesto = r.id_repuesto;  
  
UPDATE Cliente SET telefono = '3001122334'  
WHERE id_cliente = 1;  
  
UPDATE Servicio SET costo = costo + 10000  
WHERE id_servicio = 301;  
  
DELETE FROM Tecnico WHERE id_tecnico = 201;  
  
DELETE FROM Dispositivo WHERE id_dispositivo = 100;  
  
SELECT LOWER(nombre) AS cliente_minuscula  
FROM Cliente;  
  
SELECT CONCAT(tipo, ' - ', marca) AS dispositivo  
FROM Dispositivo;  
  
ALTER TABLE Cliente ADD email VARCHAR(100);  
ALTER TABLE Tecnico DROP COLUMN especialidad;  
TRUNCATE TABLE Servicio_Repuesto;  
TRUNCATE TABLE Cliente;  
DROP TABLE Dispositivo_Servicio;  
DROP TABLE Tecnico;
```



CREATE DATABASE Escuela;

USE Escuela;

```
CREATE TABLE Estudiante (
    id_estudiante INT PRIMARY KEY,
    nombre VARCHAR(100),
    fecha_nacimiento DATE
);
```

```
CREATE TABLE Profesor (
    id_profesor INT PRIMARY KEY,
    nombre VARCHAR(100),
    especialidad VARCHAR(50)
);
```

```
CREATE TABLE Curso (
    id_curso INT PRIMARY KEY,
    nombre VARCHAR(100),
    descripcion TEXT
);
```

```
CREATE TABLE Grado (
    id_grado INT PRIMARY KEY,
    nombre VARCHAR(20),
    nivel INT
);
```

```
CREATE TABLE Evaluacion (
    id_evaluacion INT PRIMARY KEY,
    id_estudiante INT,
    id_curso INT,
    nota DECIMAL (4,2),
    -
```

Nombre:

Fecha:

Profesor:

Materia:

Institución:

Curso:

Nota:

FK (id_estudiante) REFERENCES Estudiante (id_estudiante),
 FK (id_curso) REFERENCES Curso (id_curso)
);

CREATE TABLE Estudiante_Curso (
 id_estudiante INT,
 id_curso INT,
 fecha_inscripción DATE,
 PRIMARY KEY (id_estudiante, id_curso),
 FK (id_estudiante) REFERENCES Estudiante (id_estudiante),
 FK (id_curso) REFERENCES Curso (id_curso)
);

CREATE TABLE Profesor_Curso (
 id_profesor INT,
 id_curso INT,
 periodo VARCHAR(20),
 PRIMARY KEY (id_profesor, id_curso),
 FK (id_profesor) REFERENCES Profesor (id_profesor),
 FK (id_curso) REFERENCES Curso (id_curso)
);

INSERT INTO Estudiante VALUES
 (1, 'Mariana Perez', '2010-03-15'),
 (2, 'Juan Rodriguez', '2009-08-20');

INSERT INTO Profesor VALUES
 (10, 'Carlos Gomez', 'Matematicas'),
 (11, 'Laura Sanchez', 'Lengua Castellana');

INSERT INTO Curso VALUES
 (100, 'Algebra', 'Curso de ecuaciones algebraicas'),
 (101, 'Literatura', 'Obras literarias clasicas y modernas');

INSERT INTO Grado VALUES

(1, 'Sexto', 6),
 (2, 'Septimo', 7);

INSERT INTO Evaluacion VALUES
 (1000, 1, 100, 4.5),
 (1001, 2, 101, 3.8);

INSERT INTO Estudiante_Curso VALUES
 (1, 100, '2025-01-10'),
 (2, 101, '2025-01-12');

INSERT INTO Profesor_Curso VALUES
 (10, 100, '2025-1'),
 (11, 101, '2025-1');

```
SELECT nombre
FROM Curso
WHERE id_curso IN(
    SELECT id_curso
    FROM Evaluacion
    WHERE nota > 4.0 );
SELECT nombre FROM Profesor
WHERE id_profesor IN(
    SELECT id_profesor
    FROM Profesor_Curso );
SELECT p.nombre AS profesor, c.nombre AS curso
FROM Profesor p
JOIN Profesor_Curso pc ON p.id_profesor = pc.id_profesor
JOIN Curso c ON pc.id_curso = c.id_curso;
SELECT e.nombre AS estudiante, c.nombre AS curso, ev.nota
FROM Evaluacion ev
JOIN Estudiante e ON ev.id_estudiante = e.id_estudiante
JOIN Curso c ON ev.id_curso = c.id_curso;
UPDATE Evaluacion SET nota = 4.8
WHERE id_evaluacion = 1000;
UPDATE Estudiante SET nombre = 'Mariana P. Perez'
WHERE id_estudiante = 1;
DELETE FROM Evaluacion WHERE id_evaluacion = 1001;
DELETE FROM Grado WHERE id_grado = 2;
SELECT UPPER(nombre) AS profesor_mayus
FROM Profesor;
SELECT CONCAT(nombre, '(nacido:', fecha_nacimiento)
AS info FROM Estudiante;
ALTER TABLE Profesor ADD correo VARCHAR(100);
ALTER TABLE Curso DROP COLUMN descripcion;
TRUNCATE TABLE Evaluacion;
TRUNCATE TABLE Estudiante_Curso;
DROP TABLE Grado;
DROP TABLE Profesor;
```

Nombre:

Fecha:

Profesor:

Materia:

Institución:

Curso:

Nota:

1 MER

Tabla Cliente

Campo

Tipo de Dato

Descripción

id_cliente

INT

Identificador unico del cliente.

nombre

VARCHAR(100)

Nombre del Cliente

telefono

VARCHAR(20)

Número de telefono del cliente

Tabla Vehiculo

id_vehiculo

INT

Identificador unico del vehiculo

id_cliente

INT

Referencia de la tabla cliente

placa

VARCHAR(10)

Placa del vehiculo

marca

VARCHAR(50)

Marca del vehiculo

Tabla Taller

id_taller

INT

Identificador unico del taller

nombre_taller

VARCHAR(100)

Nombre del taller

direccion

VARCHAR(200)

Direccion del taller

Tabla Vehiculo-Taller

id_vehiculo

INT

Referencia de la tabla vehiculo

id_taller

INT

Referencia de la tabla taller

fecha_ingreso

DATE

Fecha de ingreso del vehiculo

fecha_salida

DATE

Fecha de salida del vehiculo

Tabla Servicio

id_servicio

INT

Identificador unico del servicio

descripcion

VARCHAR(200)

Descripcion del servicio

costo

DECIMAL(10,2)

Costo del servicio

Tabla Repuesto

Campo	Tipo de Dato	Descripción
id_reuesto	INT	Identificador único de repuesto
nombre	VARCHAR(100)	Nombre del repuesto
precio	DECIMAL(10,2)	Precio del repuesto

Tabla Servicio_Repuesto

id_servicio	INT	Referencia de la tabla servicio
id_reuesto	INT	Referencia de la tabla repuesto
cantidad	INT	Cantidad de repuestos

2 MER

Tabla Socio

id_socio	INT	Identificador único del socio
nombre	VARCHAR(50)	Nombre del socio
apellido	VARCHAR(50)	Apellido del socio

Tabla Disciplina

id_disciplina	INT	Identificador único de disciplina
nombre	VARCHAR(50)	Nombre de la disciplina
categoria	VARCHAR(30)	Categoría de la disciplina

Tabla Monitor

id_monitor	INT	Identificador único del monitor
nombre	VARCHAR(50)	Nombre del monitor
especialidad	VARCHAR(50)	Especialidad del monitor

Tabla Sala

id_sala	INT	Identificador único de la sala
nombre	VARCHAR(50)	Nombre de la sala
capacidad	INT	Capacidad de la sala

Tabla Socio-Disciplina

Campo	Tipo de Dato	Descripción
id_socio	INT	Referencia de la tabla socio
id_disciplina	INT	Referencia de la tabla disciplina
fecha_inicio	DATE	Fecha inicio del socio

Tabla Monitor-Disciplina

id_monitor	INT	Referencia de la tabla monitor
id_disciplina	INT	Referencia de la tabla disciplina
fecha_asignacion	DATE	Fecha de asignación del monitor

Tabla Reserva

id_reserva	INT	Identificador único de reserva
id_sala	INT	Referencia de la tabla sala
id_disciplina	INT	Referencia de la tabla disciplina
fecha_reserva	DATE	Fecha de reserva
	BMER	

Tabla Adoptante

id_adoptante	INT	Identificador único del adoptante
nombre	VARCHAR(100)	Nombre del adoptante
documento	VARCHAR(20)	Documento del adoptante

Tabla Mascota

id_mascota	INT	Identificador único de mascota
nombre	VARCHAR(50)	Nombre de la mascota
especie	VARCHAR(50)	Especie de la mascota
id_refugio	INT	Referencia de la tabla refugio

Tabla Adoptante-Mascota

id_adoptante	INT	Referencia de la tabla adoptante
id_mascota	INT	Referencia de la tabla mascota
fecha_adopcion	DATE	Fecha de adopción de la mascota

Tabla Veterinario

Campo , Tipo de Dato , Descripción

id_veterinario	INT	Identificador único del veterinario
nombre	VARCHAR(100)	Nombre del veterinario
especialidad	VARCHAR(50)	Especialidad del veterinario

Tabla Refugio

id_refugio	INT	Identificador único de refugio
nombre	VARCHAR(100)	Nombre del refugio
direccion	VARCHAR(100)	Direccion del refugio

Tabla Mascota-Veterinario

id_mascota	INT	Referencia de la tabla mascota
id_veterinario	INT	Referencia de la tabla veterinario
fecha_revision	DATE	Revision de la mascota

Tabla Seguimiento

id_seguimiento	INT	Identificador único de seguimiento
id_mascota	INT	Referencia de la tabla mascota
nota	VARCHAR(100)	Nota del seguimiento

4 MER

Tabla Cliente

id_cliente	INT	Identificador único de cliente
nombre	VARCHAR(100)	Nombre del cliente
correo	VARCHAR(100)	Correo del cliente

Tabla Producto

id_producto	INT	Identificador único del producto
nombre	VARCHAR(100)	Nombre del producto
precio	DECIMAL(10, 2)	Precio del producto

Nombre:

Fecha:

Profesor:

Materia:

Institución:

Curso:

Nota:

Tabla Empleado

Campo	Tipo de Dato	Descripción
id_empleado	INT	Identificador único del empleado
nombre	VARCHAR(100)	Nombre del empleado
cargo	VARCHAR(50)	Cargo que cumple el empleado

Tabla Factura

id_factura	INT	Identificador único de factura
id_cliente	INT	Referencia de la tabla cliente
id_empleado	INT	Referencia de la tabla empleado
fecha	DATE	Día de la factura

Tabla Proveedor

id_proveedor	INT	Identificador único de proveedor
nombre	VARCHAR(100)	Nombre del proveedor
contacto	VARCHAR(100)	Contacto del proveedor

Tabla Factura_Producto

id_factura	INT	Referencia de la tabla factura
id_producto	INT	Referencia de la tabla producto
cantidad	INT	Cantidad de productos

Tabla Producto_Proveedor

id_producto	INT	Referencia de la tabla producto
id_proveedor	INT	Referencia de la tabla proveedor

5 MER

Tabla Usuario

id_usuario	INT	Identificador único del usuario
nombre	VARCHAR(100)	Nombre del usuario
correo	VARCHAR(100)	Correo del usuario

Tabla Perfil

Campo Tipo de Dato Descripción

id_perfil	INT	Identificador único del perfil
id_usuario	INT	Referencia de la tabla usuario
nombre	VARCHAR(100)	Nombre del perfil

Tabla Pelicula

id_pelicula	INT	Identificador único de la pelicula
titulo	VARCHAR(100)	Título de la pelicula
duracion	INT	Duración de la pelicula

Tabla Categoría

id_categoria	INT	Identificador único de categoría
nombre	VARCHAR(50)	Nombre de la categoría

Tabla Dispositivo

id_dispositivo	INT	Identificador único de dispositivo
nombre	VARCHAR(100)	Nombre del dispositivo
tipo	VARCHAR(50)	Tipo del dispositivo

Tabla Pelicula_Categoría

id_pelicula	INT	Referencia de la tabla pelicula
id_categoria	INT	Referencia de la tabla categoria

Tabla Perfil_Pelicula

id_perfil	INT	Referencia de la tabla perfil
id_pelicula	INT	Referencia de la tabla pelicula
id_dispositivo	INT	Referencia de la tabla dispositivo
fecha	DATETIME	Fecha y hora de la pelicula

6 MER

Tabla Pasajero

id_pasajero	INT	Identificador único de pasajero
nombre	VARCHAR(100)	Nombre del pasajero
tarjeta_id	VARCHAR(50)	Número de tarjeta del pasajero

Tabla Ruta

Campo	Tipo de Dato	Descripción
id_ruta	INT	Identificador único de ruta
nombre	VARCHAR(100)	Nombre de la ruta
duracion	INT	Duración en minutos de la ruta

Tabla Bus

id_bus	INT	Identificador único del bus
placa	VARCHAR(10)	Placa del bus
capacidad	INT	Capacidad del bus

Tabla Conductor

id_conductor	INT	Identificador único del conductor
nombre	VARCHAR(100)	Nombre del conductor
licencia	VARCHAR(50)	Tipo de licencia del conductor

Tabla Estación

id_estacion	INT	Identificador único de estación
nombre	VARCHAR(100)	Nombre de la estación
direccion	VARCHAR(100)	Dirección de la estación

Tabla Bus-Ruta

id_bus	INT	Referencia de la tabla bus
id_ruta	INT	Referencia de la tabla ruta
hora_salida	TIME	Hora de salida del bus

Tabla Ruta-Estación

id_ruta	INT	Referencia de la tabla ruta
id_estacion	INT	Referencia de la tabla estación
orden	INT	Orden de las rutas de salida

7 MER Tabla Asistente

id_asistente	INT	Identificador único de asistente
nombre	VARCHAR(100)	Nombre del asistente
correo	VARCHAR(100)	Correo del asistente

Tabla Conferencia

Campo	Tipo de Dato	Descripción
id_conferencia	INT	Identificador único de conferencia
id_sala	INT	Referencia de la tabla sala
id_organizador	INT	Referencia de la tabla organizador
título	VARCHAR(50)	Título de la conferencia

Tabla Ponente

id_ponente	INT	Identificador único de ponente
nombre	VARCHAR(100)	Nombre del ponente
especialidad	VARCHAR(100)	Especialidad del ponente

Tabla Sala

id_sala	INT	Identificador único de la sala
nombre	VARCHAR(50)	Nombre de la sala
capacidad	INT	Capacidad de la sala

Tabla Organizador

id_organizador	INT	Identificador único de organizador
nombre	VARCHAR(100)	Nombre del organizador
contacto	VARCHAR(50)	Contacto del organizador

Tabla Asistente_Conferencia

id_asistente	INT	Referencia de la tabla asistente
id_conferencia	INT	Referencia de la tabla conferencia
asistencia	BOOLEAN	Asistencia a la conferencia

Tabla Ponente_Conferencia

id_ponente	INT	Referencia de la tabla ponente
id_conferencia	INT	Referencia de la tabla conferencia
tema	VARCHAR(100)	Tema de la conferencia

Nombre:

Fecha:

Profesor:

Materia:

Institución:

Curso:

Nota:

8 MER

Tabla Candidato

Campo Tipo de Dato Descripción

id_candidato	INT	Identificador único de candidato
nombre	VARCHAR(100)	Nombre del candidato
correo	VARCHAR(100)	Correo del candidato

Tabla Empresa

id_empresa	INT	Identificador único de empresa
nombre	VARCHAR(100)	Nombre de la empresa
sector	VARCHAR(50)	Sector de la empresa

Tabla Vacante

id_vacante	INT	Identificador único de vacante
titulo	VARCHAR(100)	Título del vacante
salario	DECIMAL(10,2)	Salario del vacante
id_empresa	INT	Referencia de la tabla empresa

Tabla Habilidad

id_habilidad	INT	Identificador único de habilidad
nombre	VARCHAR(50)	Nombre de la habilidad

Tabla Evaluacion

id_evaluacion	INT	Identificador único de evaluación
id_candidato	INT	Referencia de la tabla candidato
tipo	VARCHAR(50)	Tipo de evaluación
resultado	VARCHAR(50)	Resultado de la evaluación

Tabla Candidato-Habilidad

id_candidato	INT	Referencia de la tabla candidato
id_habilidad	INT	Referencia de la tabla habilidad
nivel	VARCHAR(20)	Nivel del candidato

Tabla Candidato-Vacante

Campo	Tipo de Dato	Descripción
id_candidato	INT	Referencia de la tabla candidato
id_vacante	INT	Referencia de la tabla vacante
fecha_aplicacion	DATE	Fecha de aplicación

9 MER

Tabla Cliente

id_cliente	INT	Identificador único de cliente
nombre	VARCHAR(100)	Nombre del cliente
telefono	VARCHAR(20)	Telefono del cliente

Tabla Dispositivo

id_dispositivo	INT	Identificador único de dispositivo
tipo	VARCHAR(50)	Tipo de dispositivo
marca	VARCHAR(50)	Marca del dispositivo
id_cliente	INT	Referencia de la tabla cliente

Tabla Técnico

id_tecnico	INT	Identificador único del técnico
nombre	VARCHAR(100)	Nombre del técnico
especialidad	VARCHAR(50)	Especialidad del técnico

Tabla Servicio

id_servicio	INT	Identificador único de servicio
descripcion	VARCHAR(100)	Descripción del servicio
costo	DECIMAL(10,2)	Costo del servicio

Tabla Repuesto

id_reuesto	INT	Identificador único de repuesto
nombre	VARCHAR(50)	Nombre del Repuesto
precio	DECIMAL(10,2)	Precio del repuesto

Tabla Dispositivo-Servicio

Campo	Tipo de dato	Descripción
id_dispositivo	INT	Referencia de la tabla dispositivo
id_servicio	INT	Referencia de la tabla servicio
id_tecnico	INT	Referencia de la tabla tecnico
fecha	DATE	Fecha del servicio
estado	VARCHAR(20)	Estado del dispositivo

Tabla Servicio-Repuesto

id_servicio	INT	Referencia de la tabla servicio
id_repuesto	INT	Referencia de la tabla repuesto
cantidad	INT	Cantidad de repuestos
	10 MER	

Tabla Estudiante

id_estudiante	INT	Identificador unico de estudiante
nombre	VARCHAR(100)	Nombre del estudiante
fecha_nacimiento	DATE	Fecha de nacimiento del estudiante

Tabla Profesor

id_profesor	INT	Identificador unico del profesor
nombre	VARCHAR(100)	Nombre del profesor
especialidad	VARCHAR(50)	Especialidad del profesor

Tabla Curso

id_curso	INT	Identificador unico del curso
nombre	VARCHAR(100)	Nombre del curso
descripcion	TEXT	Descripcion del curso

Tabla Grado

id_grado	INT	Identificador unico de grado
nombre	VARCHAR(20)	Nombre del grado
nivel	INT	Nivel del grado

Tabla Evaluacion

Campo	Tipo de Dato	Descripción
id_evaluacion	INT	Identificador unico de evaluacion
id_estudiante	INT	Referencia de la tabla estudiante
id_curso	INT	Referencia de la tabla curso
nota	DECIMAL(4,2)	Nota de la evaluacion

Tabla Estudiante_Curso

id_estudiante	INT	Referencia de la tabla estudiante
id_curso	INT	Referencia de la tabla curso
fecha_inscripcion	DATE	Fecha inscripcion del estudiante

Tabla Profesor_Curso

id_profesor	INT	Referencia de la tabla profesor
id_curso	INT	Referencia de la tabla curso
periodo	VARCHAR(20)	Periodo en el que va el curso