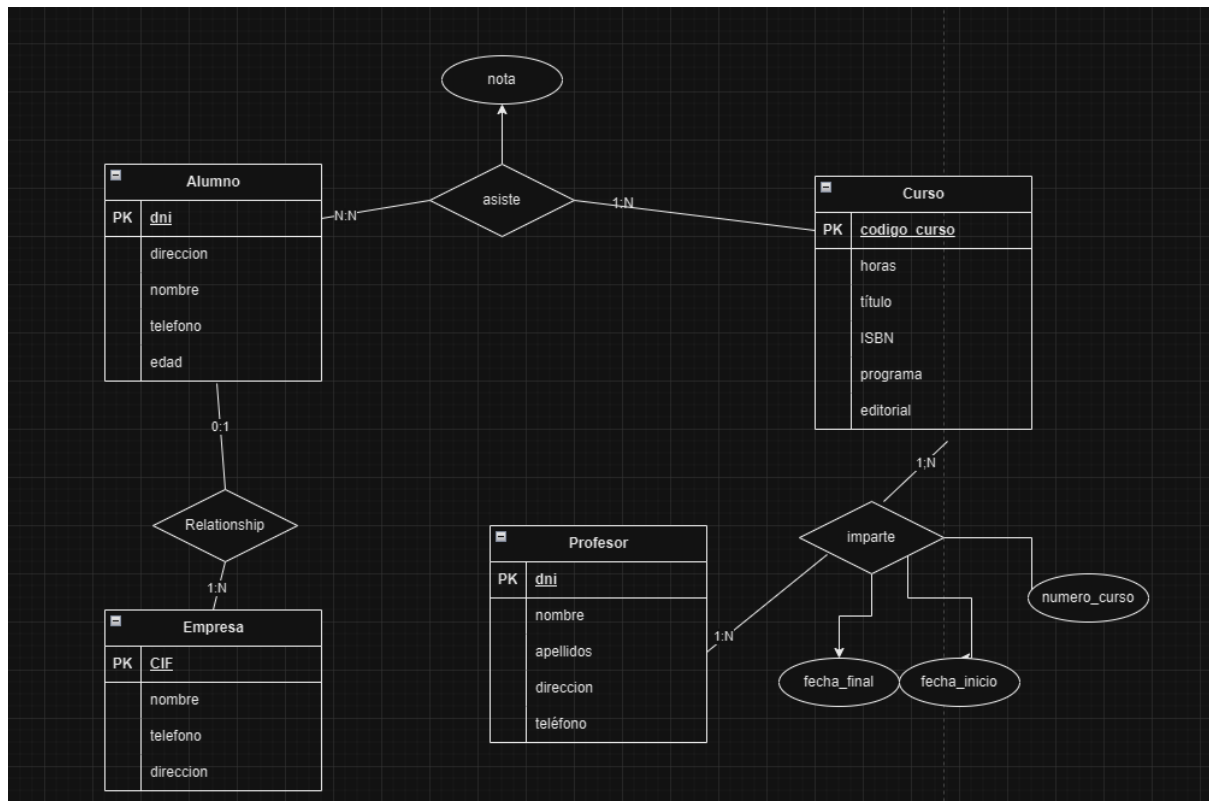


Ejercicio 2



Empresa(cif: string, nombre: string, telefono: string, direccion: string)

Alumno(dni: string, direccion: string, nombre: string, telefono: string, edad: number, empresa: string (FK))

- *empresa es FK que hace referencia a Empresa(CIF)*

Curso(codigo_curso: number, programa: string, horas: date, titulo: string, editorial: string, ISBN: string)

Asistencia(dni_alumno_FK:string, codigo_curso_FK: string, nota: number)

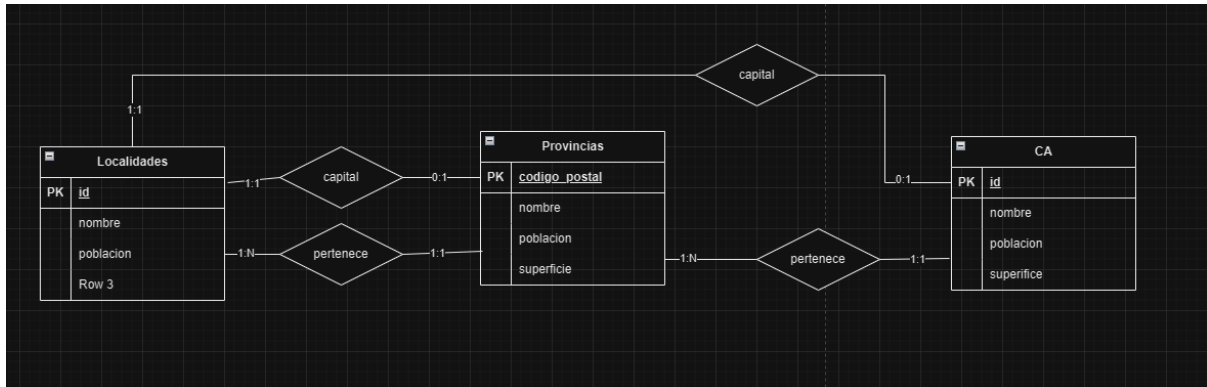
- *Dni_alumno y codigo_curso son a su vez FK que hacen referencia a Curso(codigo_curso) y Alumno(dni).*

Profesor(dni:number, nombre: string, apellidos: string, direccion: string, telefono: string)

Imparte(num_curso: number, dni_profesor_FK:string, codigo_curso_FK:number, fecha_inicio: date, fecha_final: date)

- *Dni_profesor es FK que hace referencia a professor(dni)*
- *Codigo_curso es FK que hace referencia a Curso(codigo_curso)*

Ejercicio 3



Localidades(**id**: number, nombre: string, poblacion: number)

Provincias(**codigo_postal**: number, nombre: string, poblacion: number, superficie: number, id_capital: number(FK))

- *id_capital es FK que hace referencia a Localidades(id)*

Pertenencia_localidad_provincia(id_localidad: number(FK), codigo_postal_provincia: number(FK))

- *Id_localidad es FK que hace referencia a Localidades(id) y codigo_postal_provincia es FK que hace referencia a Provincias(codigo_postal)*

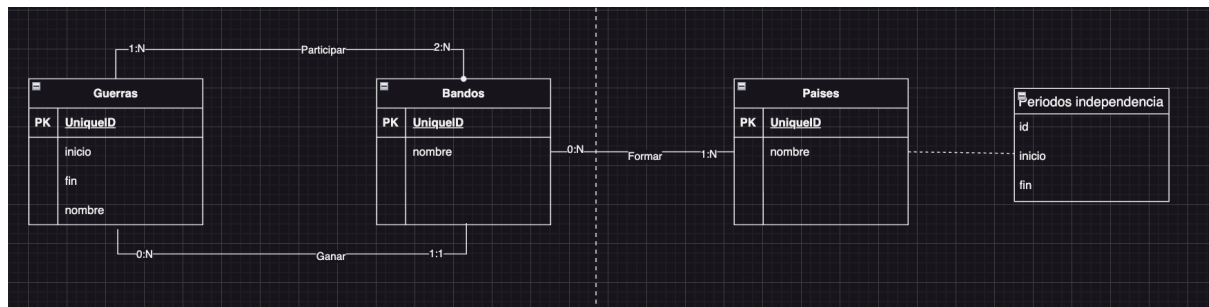
Comunidades_autonomas(**id**: number, nombre: string, poblacion: number, superficie: number, id_capital: number(FK))

- *id_capital es FK que hace referencia a Localidades(id)*

Pertenencia_provincia_CA(codigo_postal_provincia: number(FK), id_CA: number(FK))

- *codigo_postal_provincia es FK que hace referencia a Provincias(codigo_postal)*
- *id_CA es FK que hace referencia a Comunidades_autonomas(id)*

Ejercicio 4. Guerras



Bandos(**id**:number, nombre:string)

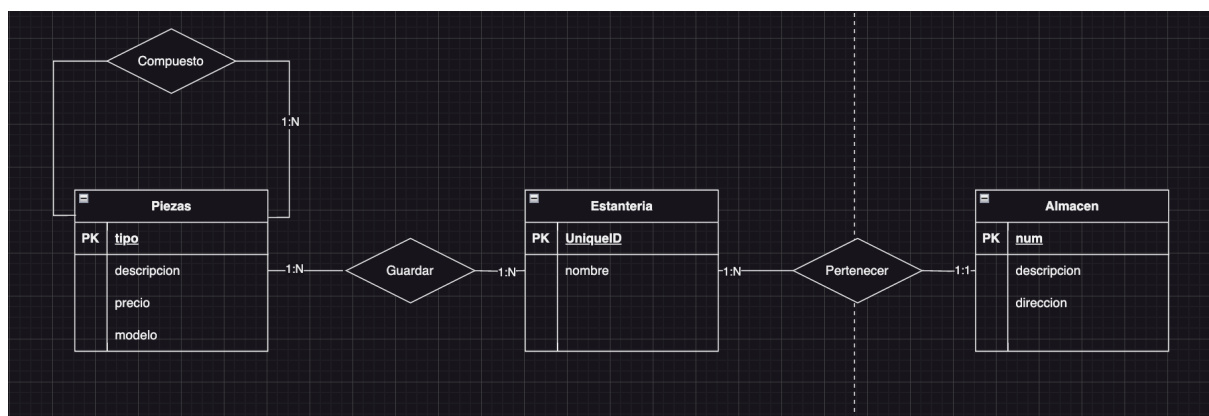
Guerras(**id**:number, inicio:number, fin:number, nombre:string, ganador:number(FK))

Participaciones(**id_guerra**:number(FK), **id_bando**:number(FK))

Países(**id**:number, nombre:string, bando:id_bando(FK))

Periodos_independencia(**id_pais**:number(FK), inicio:number, fin:number)

Ejercicio 5. Almacén v1



Pieza(**tipo**:string, descripcion:string, precio:number, modelo:string)

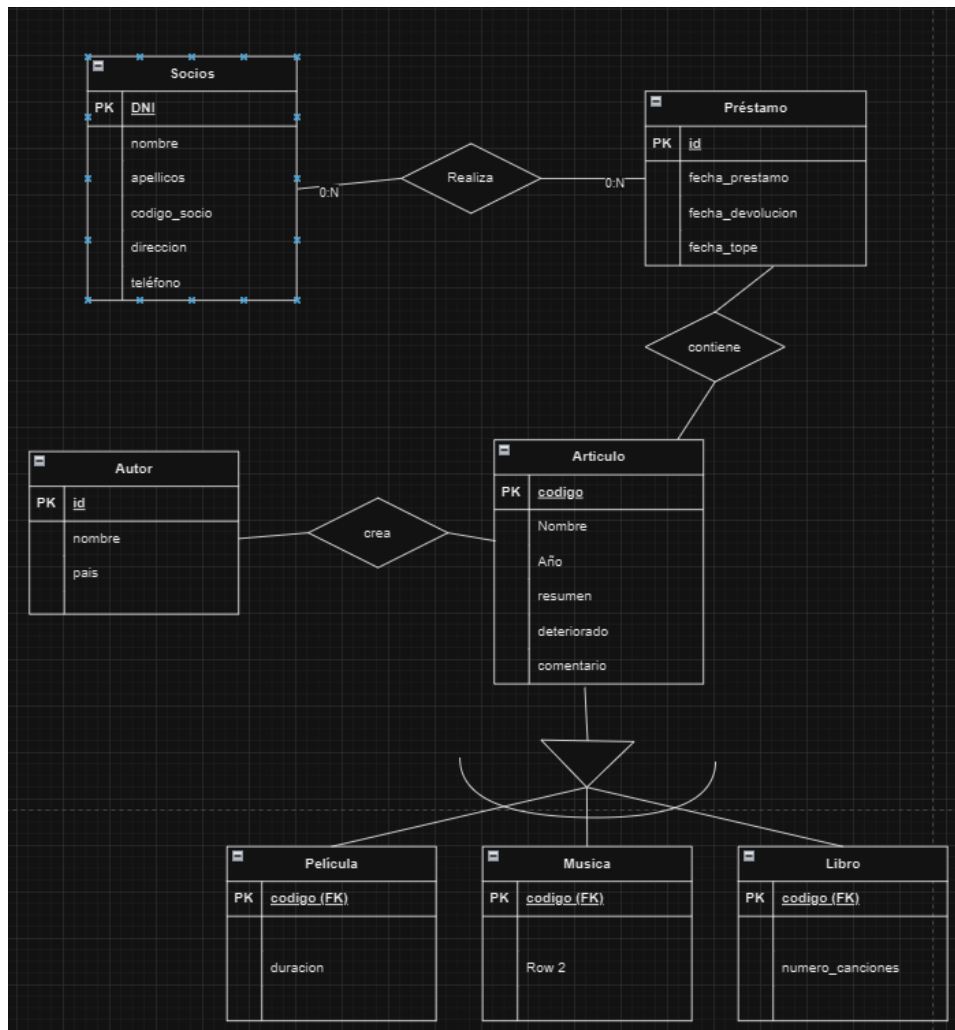
Compuesto(**pieza1**:id_pieza(FK), **pieza2**:id_pieza(FK))

Almacen(**num**:number, descripcion:string, direccion:string)

Estanteria(**id**:number, nombre:string, **id_almacen**:id_almacen(FK))

Almacen(**numero**:number, descripcion:string, direccion:string)

Ejercicio 6

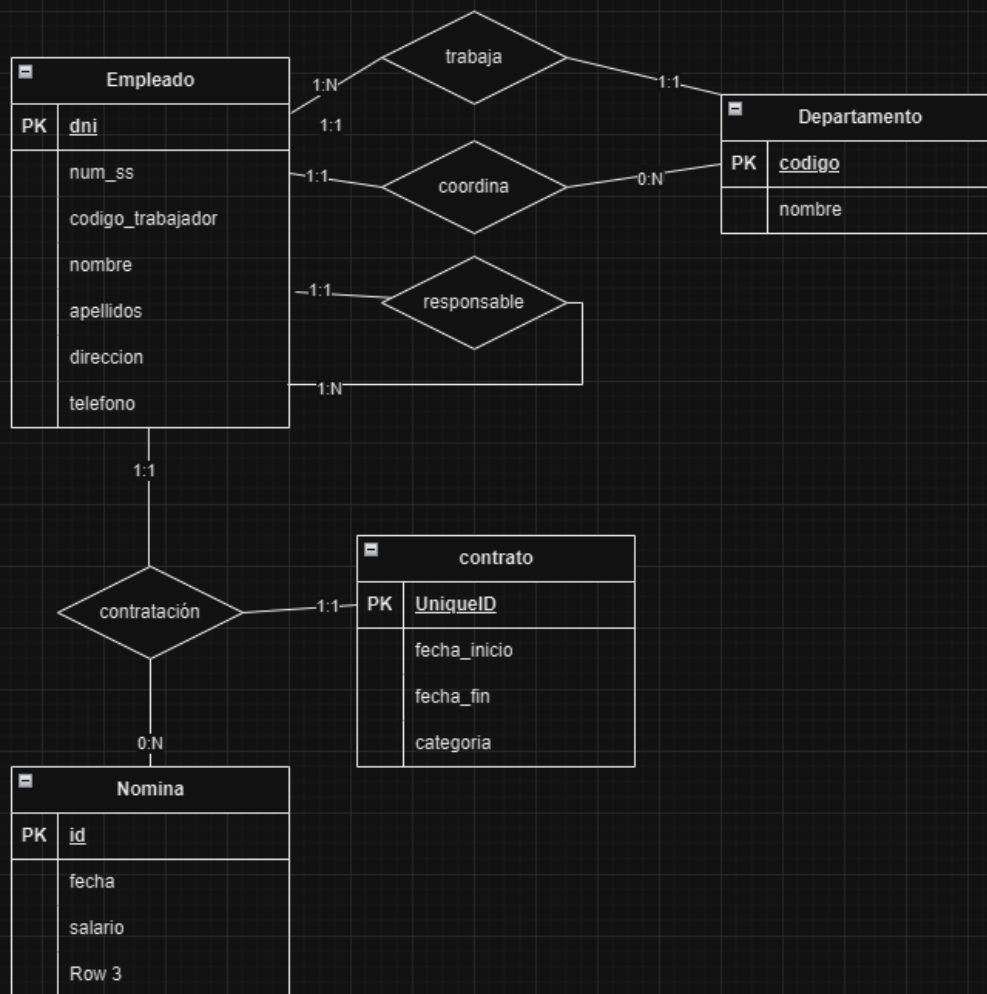


Socios(dni:string, nombre: string, apellidos: string, codigo_socio: number, direccion: string, telefono: string)

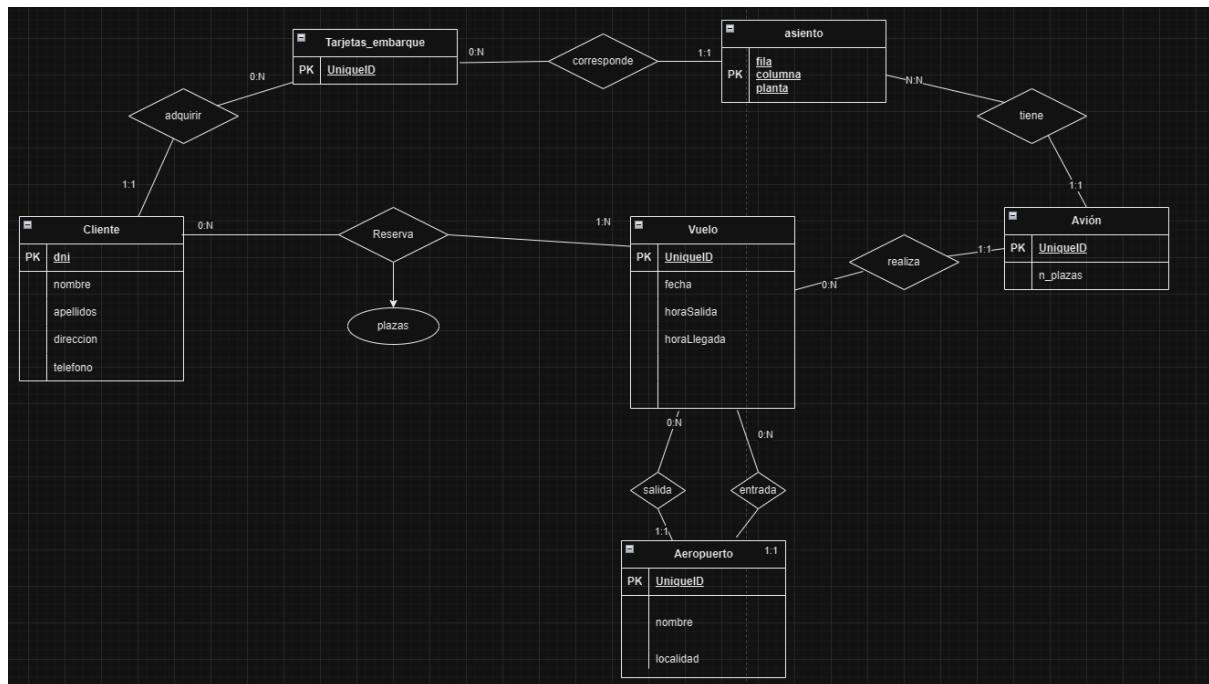
Préstamo(id : number, fecha_prestamo: date, fecha_devolucion: date, fecha_tope: date)

Realiza(dni_socio: string(FK). id prestamo: number(FK))

Ejercicio 7



Ejercicio 8. Vuelos



Cliente(**dni**: string, nombre:string, apellidos: string, direccion: string, telefono: string)

Aeropuerto(**id**: number, nombre: string, localidad: string)

Vuelo(**id**: number, fecha: date, hora_salida:date, horaLlegada:date, aeropuerto_salida_id: number(FK), aeropuerto_entrada_id:number(FK), id_avion: number(FK))

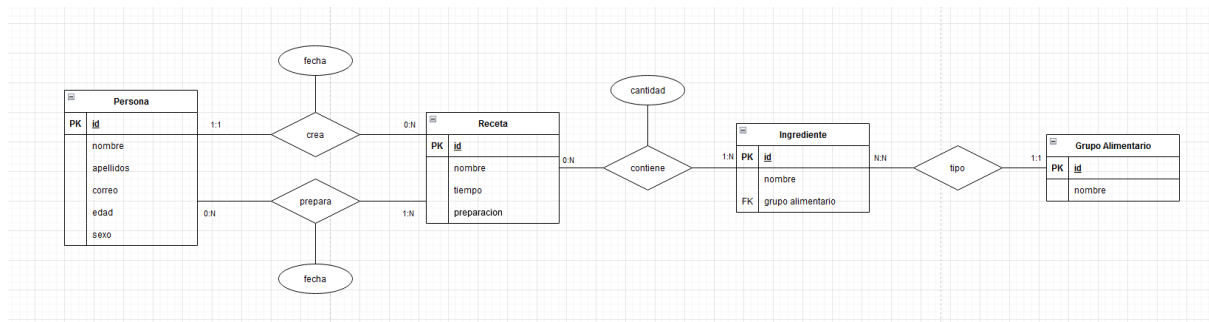
Reserva(dni_cliente:string(FK), id_vuelo:number(FK), **numero_reserva**: number, plazas: number)

Avion(**id**: number, num_plazas: number)

Asiento(**fila**: number, **columna**: number, **planta**: number, id_avion: number(FK))

Tarjetas_embarque(**id**: number, dni_cliente: string(FK), fila: number(FK), columna: number(FK), planta: number(FK), id_avion: number(FK))

Ejercicio 9. Recetas de cocina



Personas(**id**:number, nombre:string , apellidos:string, correo:string, edad:number, sexo: string)

Recetas(**id**:number, nombre:string, tiempo:number, preparacion:string, fecha:date, id_persona_FK:number)

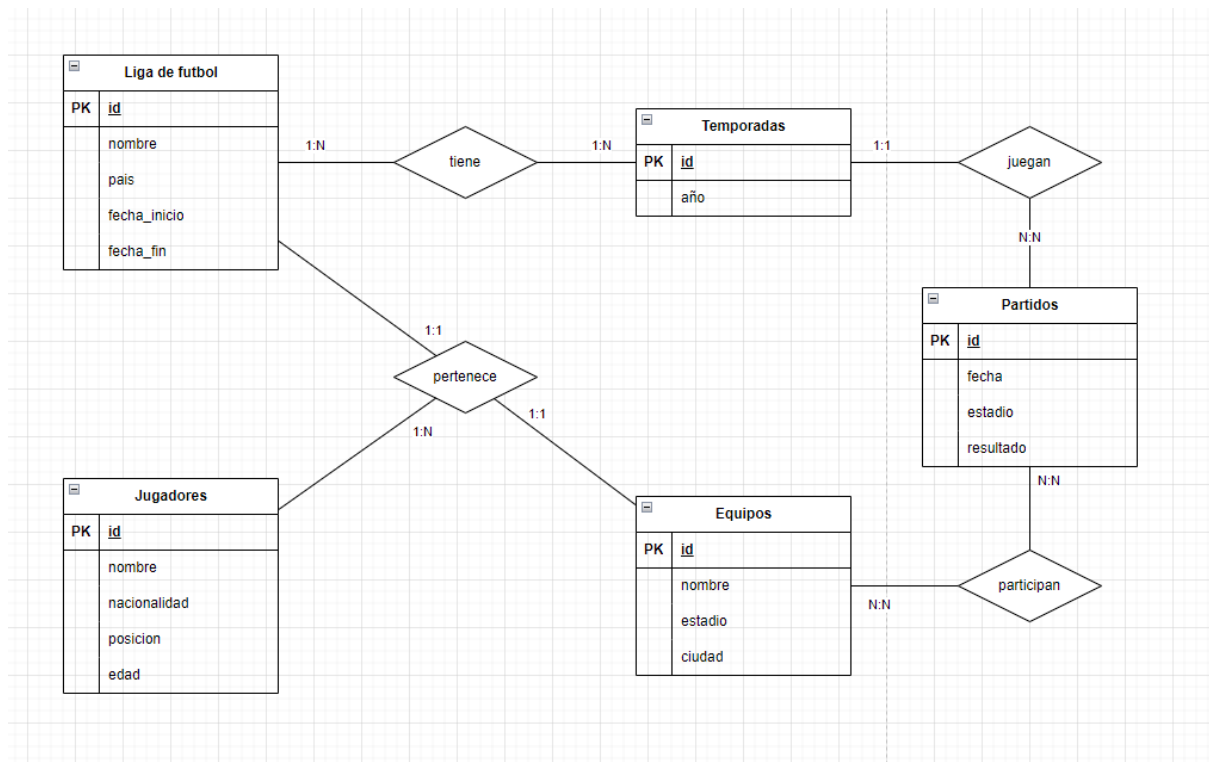
Cocineros(id_persona_FK:number, id_receta_FK:number, fecha:date)

Grupo_alimentario(**id**:number, nombre:string)

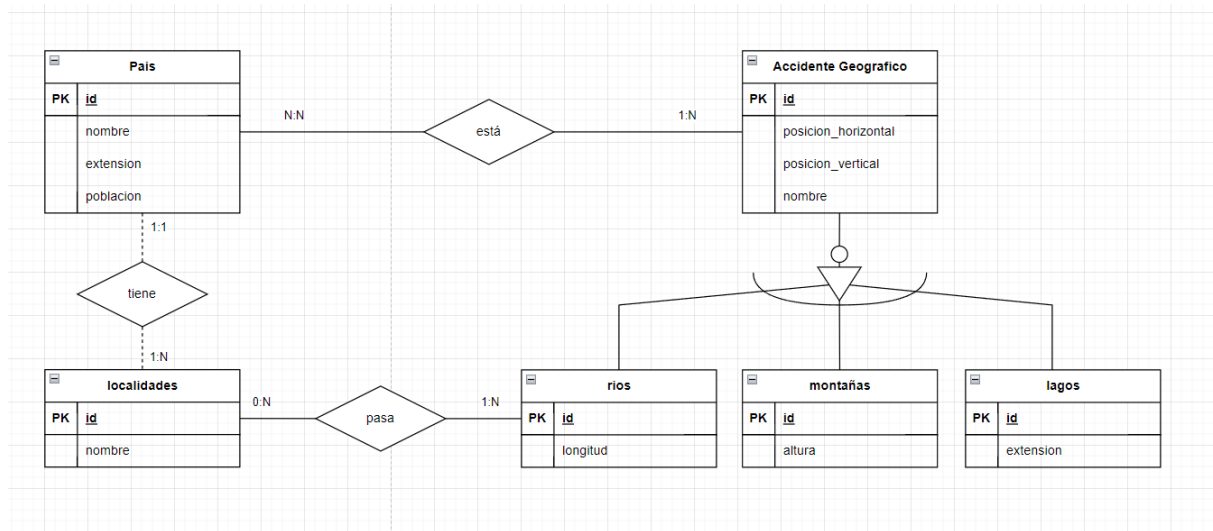
Ingredientes(**id**:number, nombre:string, id_grupo_alimentario_FK:string)

Contiene(id_receta_FK:number, id_ingrediente_FK:number, cantidad:number)

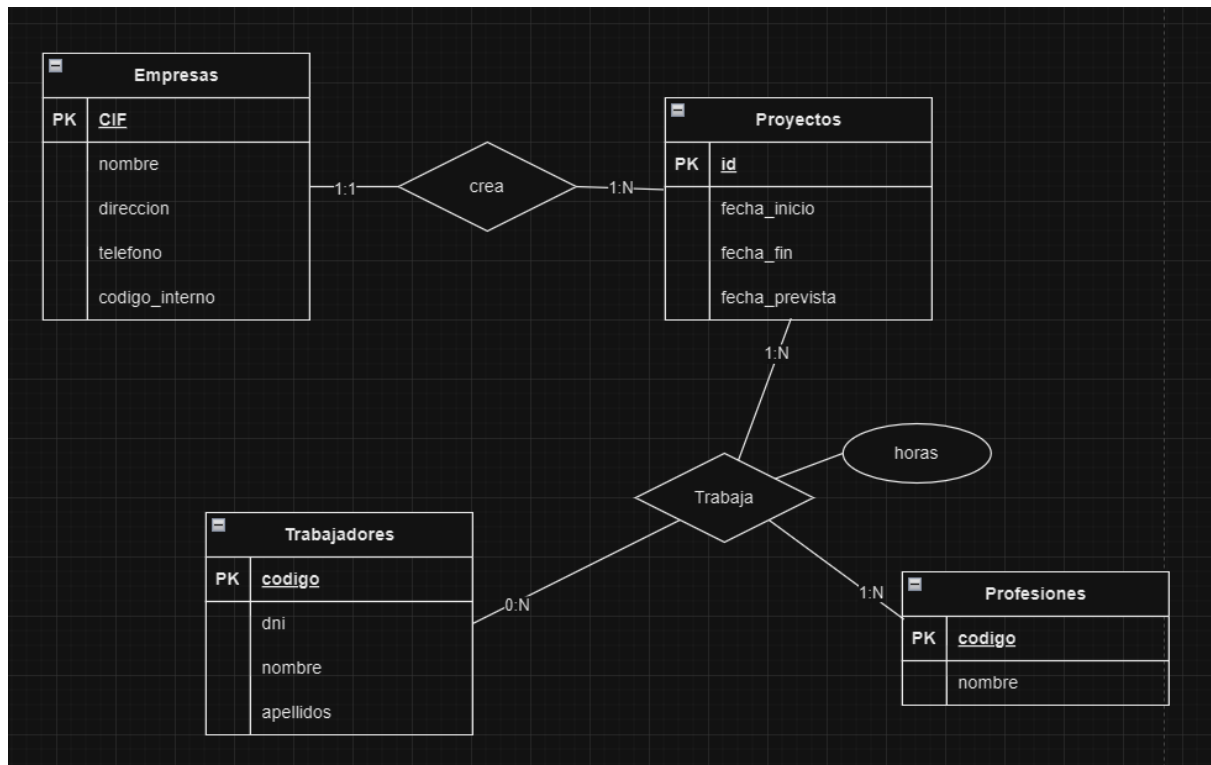
Ejercicio 10



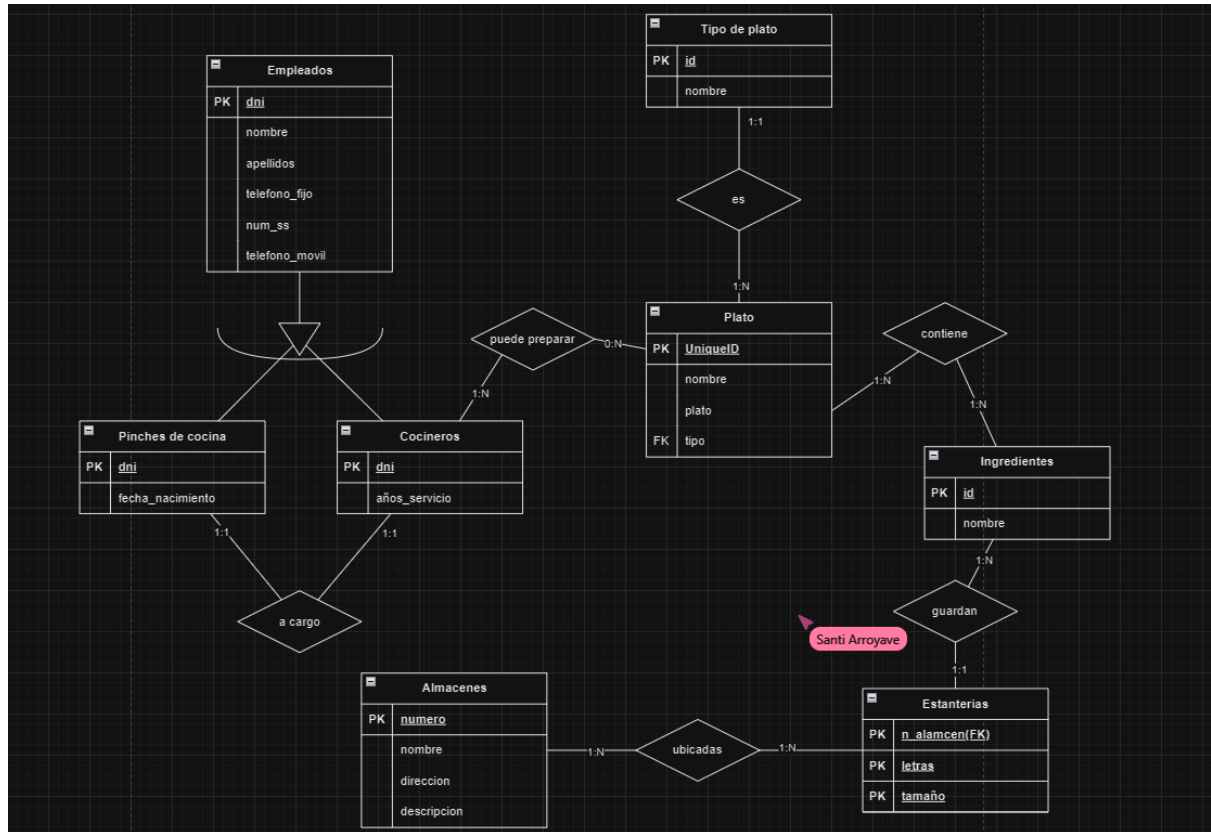
Ejercicio 11



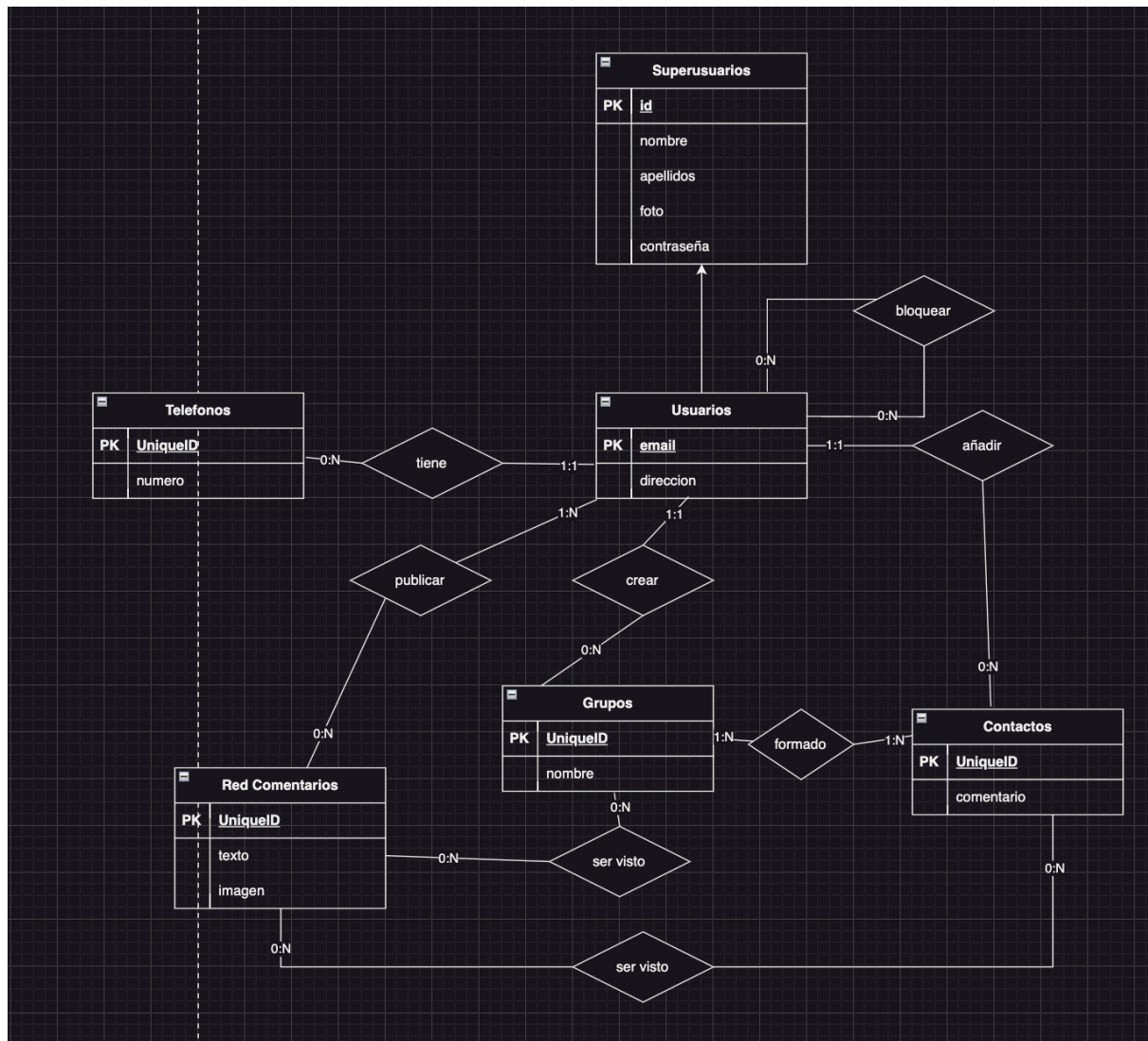
Ejercicio 12



Ejercicio 13



Ejercicio 14. Red Social



Superusuarios()

Usuarios()

Telefonos()

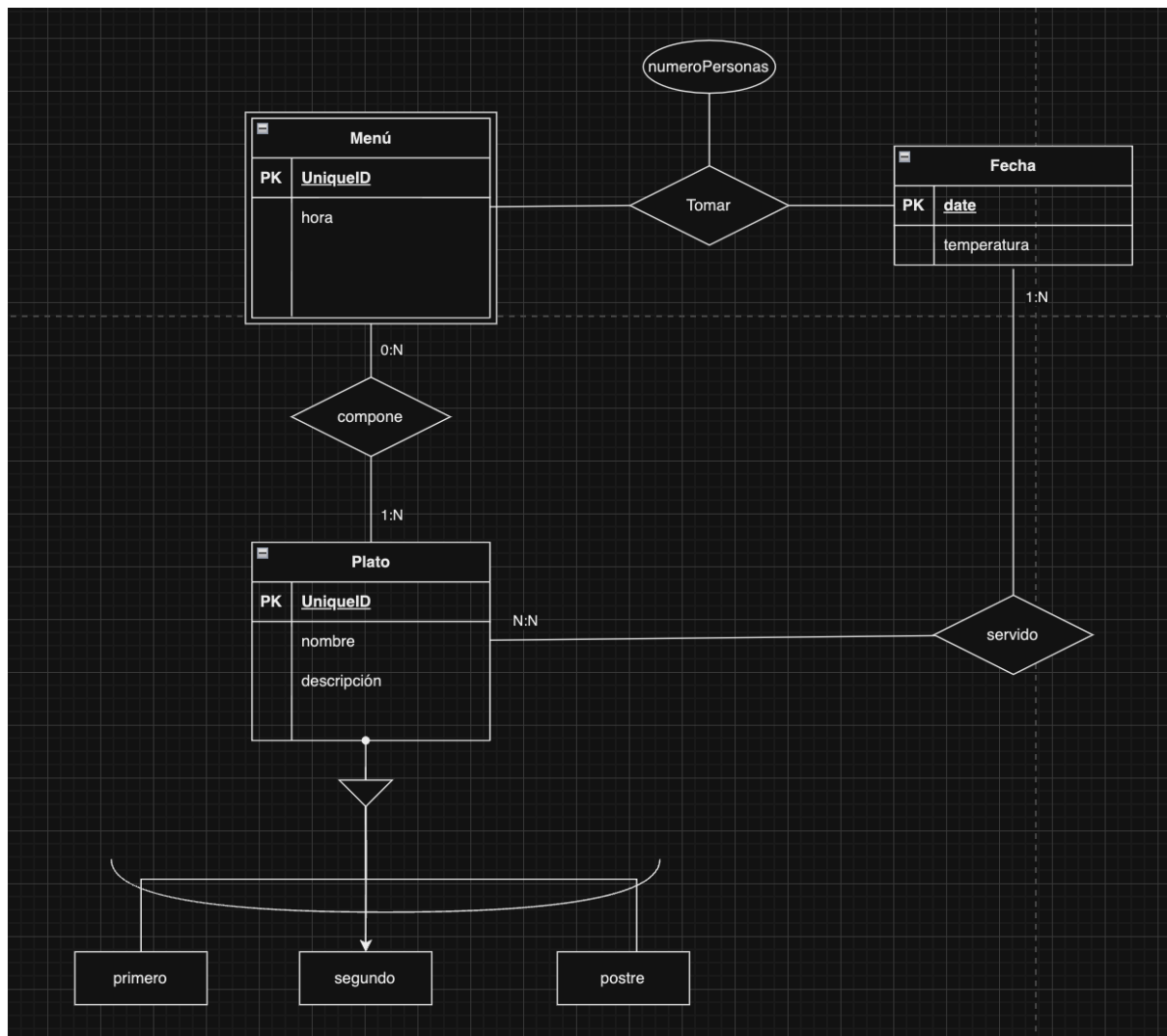
Contactos()

Grupos()

RedComentarios()

Usuario_redcomentarios()

Ejercicio 15. Menú diario



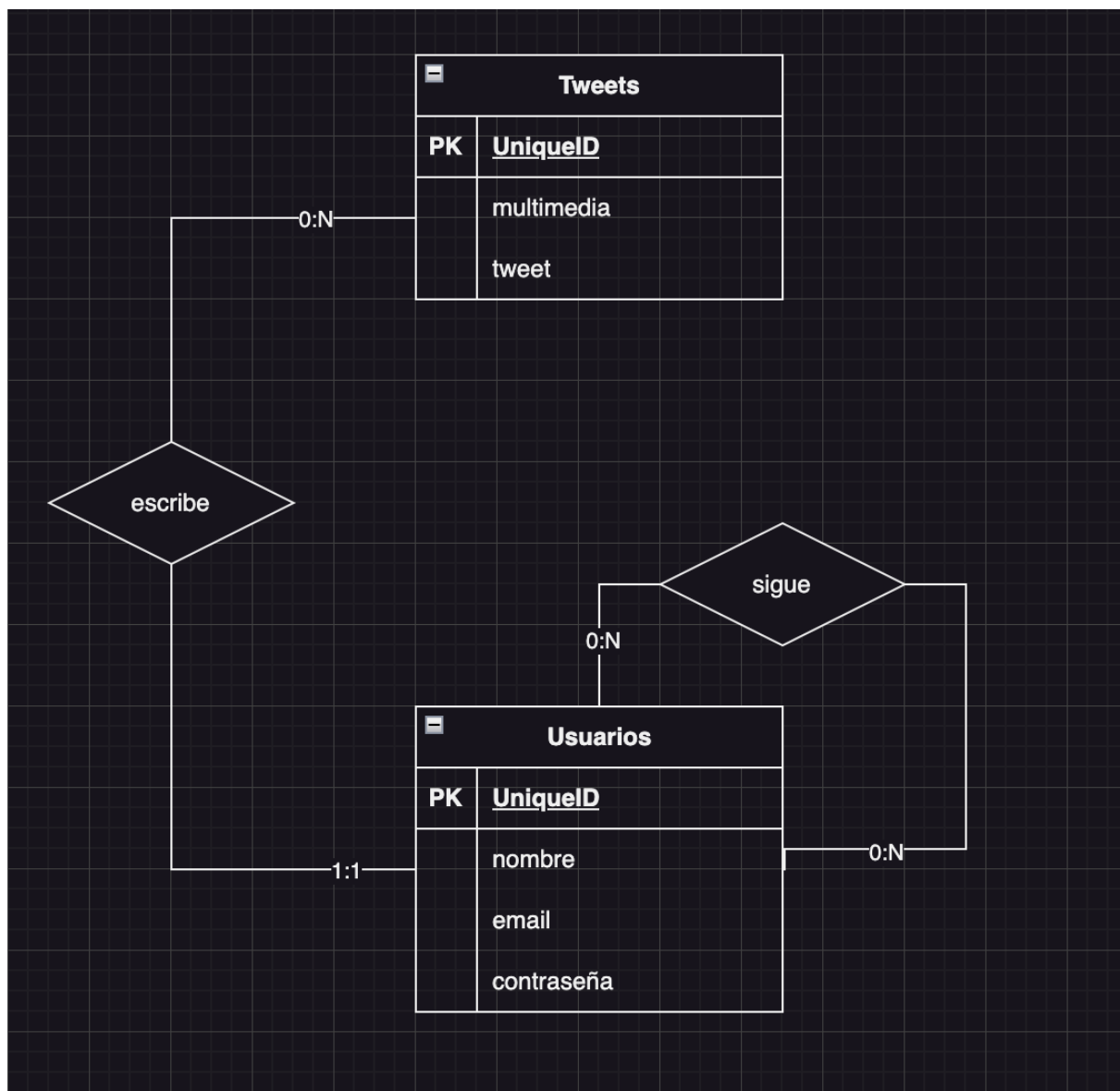
Plato(id:number,nombre:string,descripcion:string)

Primero(id_plato(FK))

Segundo(id_plato(FK))

Postre(id_plato(FK))
Menu(id:number,hora:string)
Fecha(date:date,temperatura:string)
Menu_Plato(id_menu(FK),id_plato(FK))
Menu_Fecha(id_menu(FK),date(FK),num_personas:number)
Plato_Fecha(id_plato(FK),date(FK),cantidad:number,exitoso:boolean)

Ejercicio 16. Twitter



Usuarios(id,nombre,email,contraseña)
Tweets(id,multimedia,tweet)
Seguidores(seguido:id_usuario(FK),seguidor:id_usuario(FK))
Usuario_Tweet(tweet:id_tweet(FK),usuario:id_usuario(FK))

DDL

```
create database ex16;
```

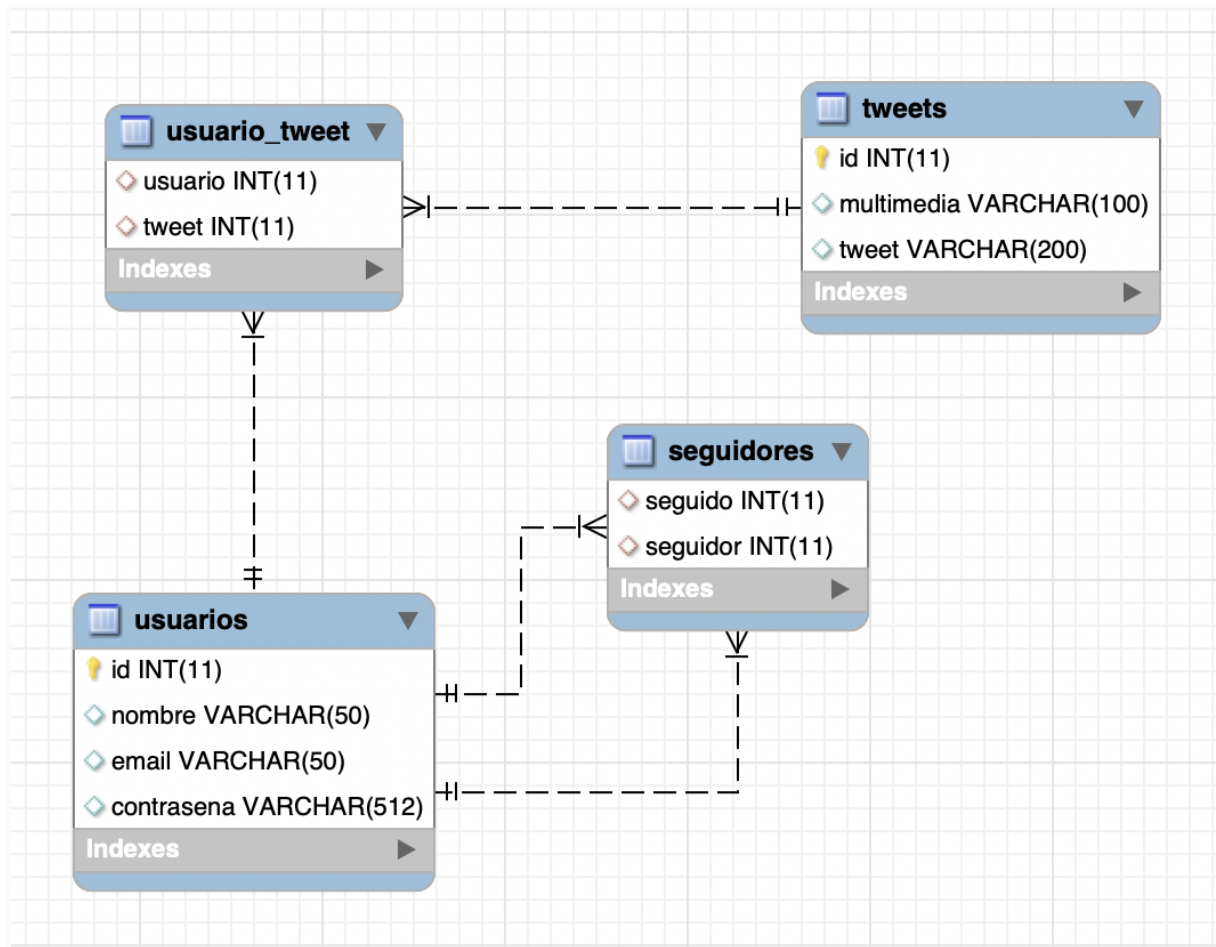
```
use ex16;
```

```
create table usuarios(  
    id int auto_increment primary key,  
    nombre varchar(50),  
    email varchar(50),  
    contrasena varchar(512)  
);
```

```
create table tweets(  
    id int auto_increment primary key,  
    multimedia varchar(100),  
    tweet varchar(200)  
);
```

```
create table seguidores(  
    seguido int,  
    seguidor int,  
    foreign key(seguido) references usuarios(id)  
    on delete cascade  
    on update cascade,  
    foreign key(seguidor) references usuarios(id)  
    on delete cascade  
    on update cascade  
);
```

```
create table usuario_tweet(  
    usuario int,  
    tweet int,  
    foreign key(usuario) references usuarios(id)  
    on delete cascade  
    on update cascade,  
    foreign key(tweet) references tweets(id)  
    on delete cascade  
    on update cascade  
);
```



DML

```

INSERT INTO usuarios (nombre, email, contraseña) VALUES
('Usuario1', 'usuario1@example.com', 'contrasena123'),
('Usuario2', 'usuario2@example.com', 'clave456'),
('Usuario3', 'usuario3@example.com', 'pass789');

```

```

INSERT INTO tweets (multimedia, tweet) VALUES
('imagen1.jpg', 'Este es el primer tweet.'),
('video1.mp4', 'Compartiendo un video interesante.'),
(NULL, 'Un tweet sin multimedia.'),
('imagen2.png', '¡Seguimos compartiendo contenido!');

```

```

INSERT INTO seguidores (seguido, seguidor) VALUES
(1, 2),
(1, 3);

```

```

INSERT INTO seguidores (seguido, seguidor) VALUES
(2, 1);

```

```

INSERT INTO seguidores (seguido, seguidor) VALUES
(3, 1);

```

```
INSERT INTO usuario_tweet (usuario, tweet) VALUES  
(1, 1),  
(1, 3);
```

```
INSERT INTO usuario_tweet (usuario, tweet) VALUES  
(2, 2),  
(2, 4);
```

```
INSERT INTO usuario_tweet (usuario, tweet) VALUES  
(3, 3),  
(3, 4);
```

Ejercicio 17

