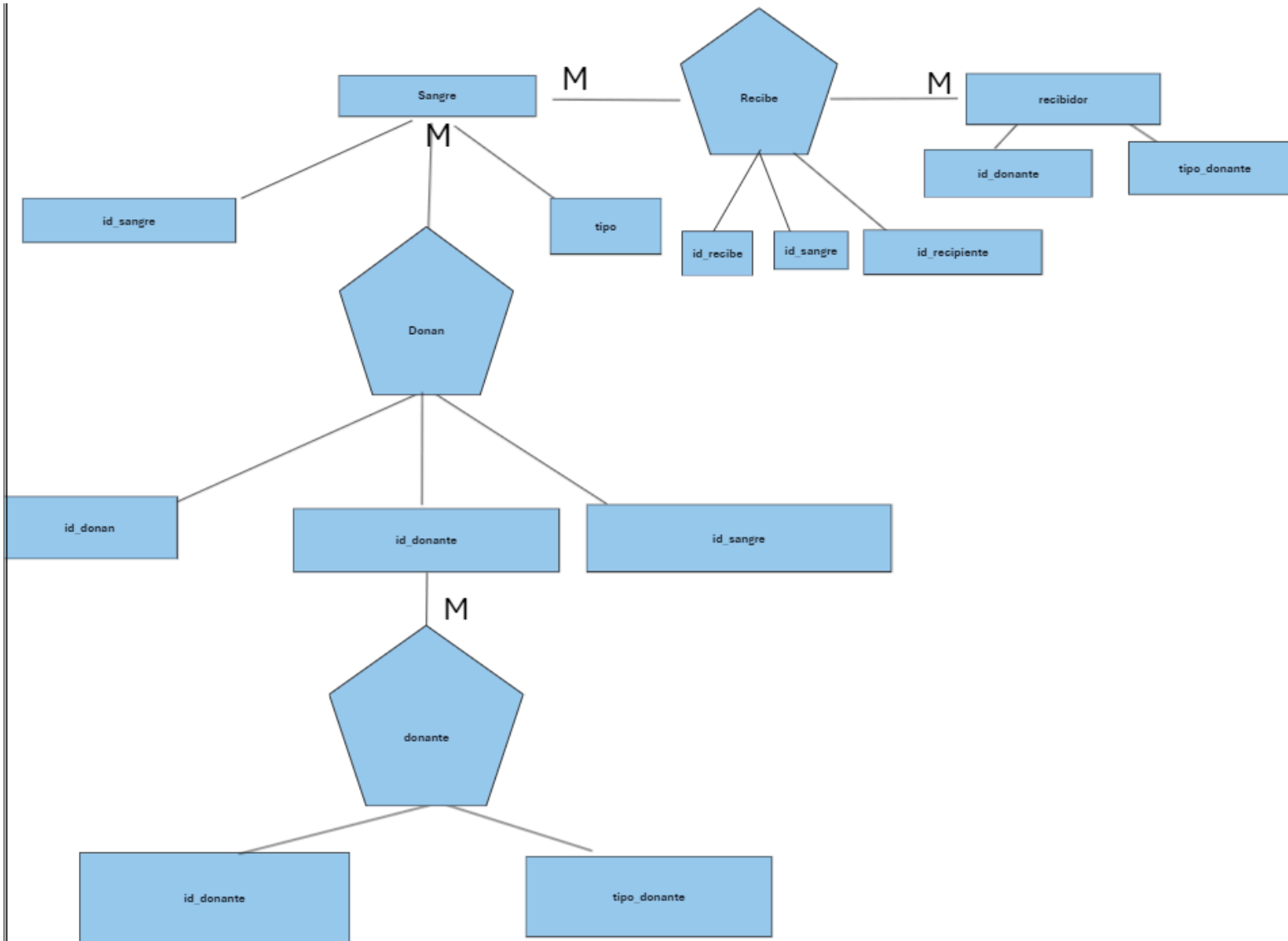


ÍNDICE

Introducción.....	3
CONCLUSIÓN.....	4

Introducción

En este ejercicio lo que vamos a hacer es relacionar varias tablas relacionado con la donación de sangre



```

create database bancoSangre;
use bancoSangre;
create table sangre (
    id_sangre int(100) primary key auto_increment,
    tipo varchar (100) not null);

create table donante(
    id_donante int(100) primary key auto_increment,
    tipo_donante varchar(100) not null
);

create table recibidor(
    id_recipiente int(100) primary key auto_increment,
    tipo_donante varchar(100) not null
);

create table recibe(
    id_recibe int(100) primary key auto_increment,
    id_sangre int(100) not null,
    id_recipiente int (100) not null,
    foreign key(id_sangre) references sangre (id_sangre),
    foreign key(id_recipiente) references recibidor (id_recipiente)
);

create table donan(
    id_donan int(100) primary key auto_increment,
    id_sangre int(100) not null,
    id_donante int(100) not null,
    foreign key(id_sangre) references sangre (id_sangre),
    foreign key(id_donante) references donante (id_donante)
);
  
```

✓ MySQL ha devuelto un conjunto de valores vacío (es decir: cero columnas). (La consulta tardó 0,0002 segundos.)

```
create table sangre ( id_sangre int(100) primary key auto_increment, tipo varchar(100) not null);
```

[Editar en línea] [Editar] [Crear código PHP]

✓ MySQL ha devuelto un conjunto de valores vacío (es decir: cero columnas). (La consulta tardó 0,0002 segundos.)

```
create table donante( id_donante int(100) primary key auto_increment, tipo_donante varchar(100) not null );
```

[Editar en línea] [Editar] [Crear código PHP]

✓ MySQL ha devuelto un conjunto de valores vacío (es decir: cero columnas). (La consulta tardó 0,0002 segundos.)

```
create table recibidor( id_recipiente int(100) primary key auto_increment, tipo_donante varchar(100) not null );
```

[Editar en línea] [Editar] [Crear código PHP]

✓ MySQL ha devuelto un conjunto de valores vacío (es decir: cero columnas). (La consulta tardó 0,0002 segundos.)

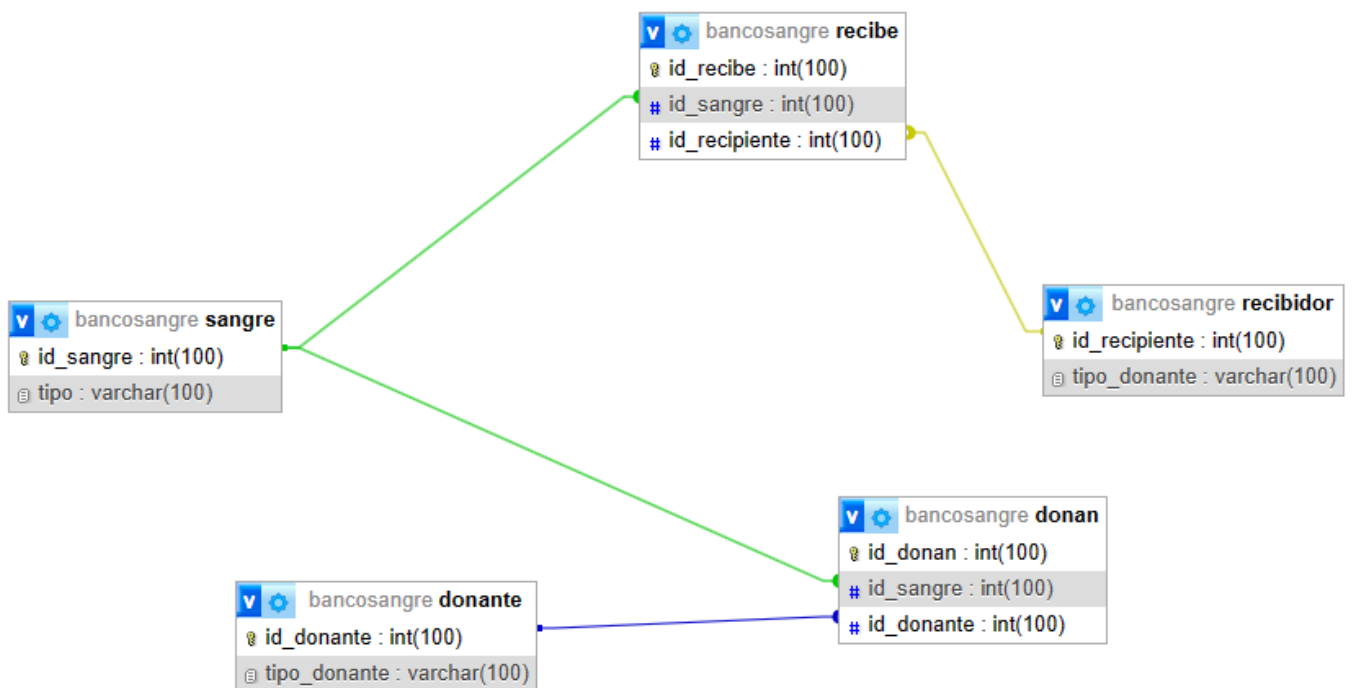
```
create table recibe( id_recibe int(100) primary key auto_increment, id_sangre int(100) not null, id_recipiente int(100) not null, foreign key(id_sangre) references sangre (id_sangre), foreign key(id_recipiente) references recibidor (id_recipiente) );
```

[Editar en línea] [Editar] [Crear código PHP]

✓ MySQL ha devuelto un conjunto de valores vacío (es decir: cero columnas). (La consulta tardó 0,0002 segundos.)

```
create table donan( id_donan int(100) primary key auto_increment, id_sangre int(100) not null, id_donante int(100) not null, foreign key(id_sangre) references sangre (id_sangre), foreign key(id_donante) references donante (id_donante) );
```

[Editar en línea] [Editar] [Crear código PHP]



```
create table donante(  
    id_donante int(100) primary key auto_increment,  
    tipo_donante varchar(100) not null,  
    nombre varchar (255) not null  
);
```

Ahora Agregamos los nombres en el donante y en el receptor

```
create table receptor(  
    id_receptor int(100) primary key auto_increment,  
    tipo_donante varchar(100) not null,  
    nombre varchar(100) not null  
);
```

Y aquí agregamos para una mayor información pues añadimos la fecha de donación y fecha de transfusión + la dirección.

```
create table donan(  
    id_donan int(100) primary key auto_increment,  
    id_sangre int(100) not null,  
    id_donante int(100) not null,  
    fecha_donacion date,  
    fecha_transfusion,  
    direccion varchar (255) not null,  
    foreign key(id_sangre) references sangre (id_sangre),  
    foreign key(id_donante) references donante (id_donante)  
);
```

CONCLUSIÓN

Esta actividad me ha ayudado a entender mejor la entidad relacion que hay entre tablas