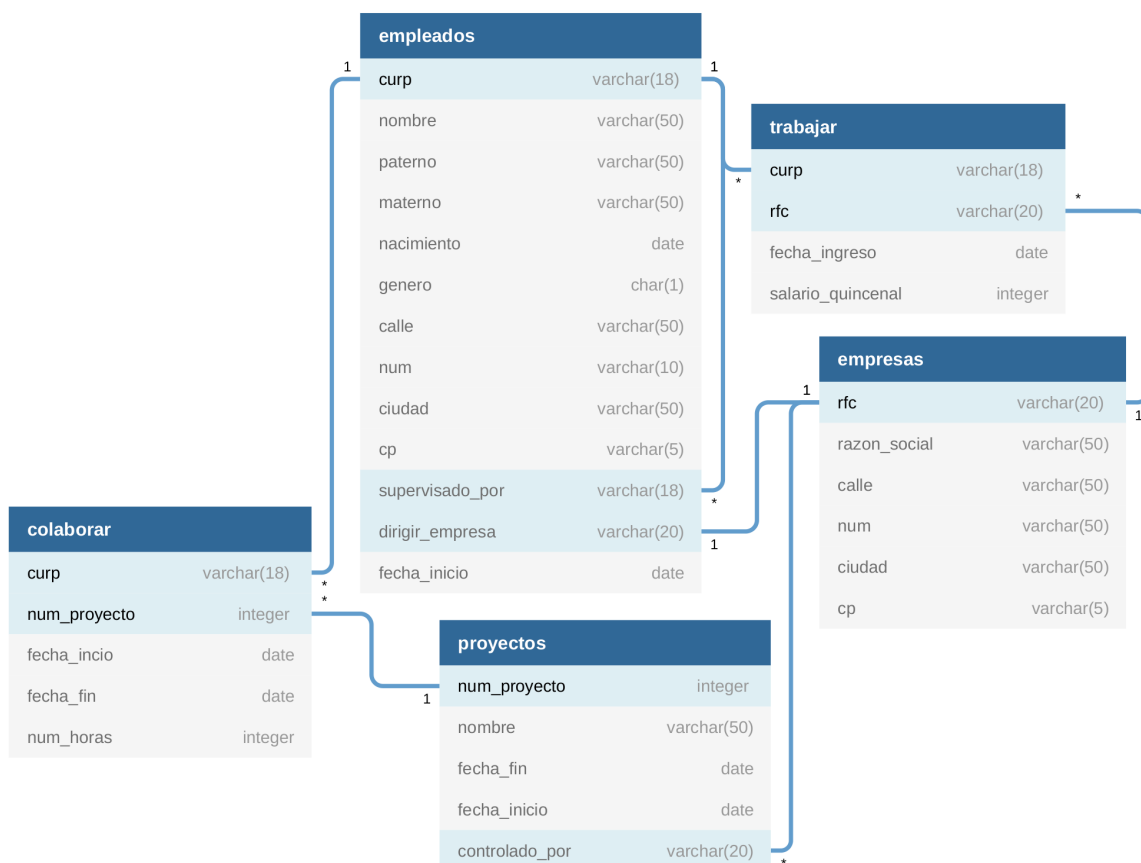


## Tarea 6 SQL

0.1. Traduce el modelo E-R a su correspondiente modelo relacional, indicando claramente las llaves primarias y foráneas. No incluyas relaciones redundantes.



Para la tabla de empleados se utilizó el curp como llave primaria tal y como viene en el modelo entidad-relación, para el atributo nombreC se eliminó y se puso directamente nombre, paterno y materno. También se añadieron a esta tabla los atributos dirigir\_empresa y fecha\_inicio para la relación de dirigir ya que necesitamos saber qué empleados dirigen que empresa, en caso de que no dirijan una empresa estos campos se quedan en null. Por último también se añadió el campo de supervisado\_por para la relación supervisar que hace referencia al supervisor del empleado.

Para la tabla de empresas también tomamos como llave primaria la que ya venía en el modelo entidad relación y tomamos los atributos de dirección para ponerlos directamente en la tabla.

Para la tabla trabajar, formamos una llave primaria con curp y rfc, para representar que el usuario con ese curp trabaja en la empresa con ese rfc, esto para que pueda haber empleados que trabajan en más de una empresa. Los atributos fecha\_ingreso y salario\_quincenal los pusimos directamente en la tabla.

Para la tabla proyectos usamos como llave primaria la que ya venía, que es num\_proyecto y para la relación controlar le añadimos una columna llamada controlada\_por en la que guardamos el rfc de la empresa a la que pertenece, esto por que cada proyecto debe estar limitado a una sola empresa. Los demás atributos de la entidad los pusimos directamente como columnas.

Para la tabla colaborar, pusimos directamente los atributos de la relación como columnas y además pusimos la columna curp y num\_proyecto para formar su llave primaria y para poder representar que un empleado puede estar en diversos proyectos y a su vez que un proyecto puede tener muchos empleados.

## **0.2. Indica para la política de mantenimiento de llaves foráneas, qué ventajas y desventajas que tienen las políticas de establecimiento de nulos y cascada.**

Se utilizó:

1. ON DELETE CASCADE NOT NULL
2. ON DELETE NO ACTION NOT NULL

como políticas de mantenimiento, además a cada llave foráneas tiene integridad referencial.

### **1. Establecimiento de nulos:**

#### **Ventajas:**

- ON DELETE SET NULL

Al eliminar una fila con una llave a la que hacen referencia las llaves foráneas de las filas existentes de otras tablas, las filas que dependan de esta llave foránea serán NULL, por tanto se necesita que se acepte NULL en esos valores.

- ON UPDATE SET NULL

Es el caso análogo al de arriba, solo que al actualizarse en vez de eliminar, el comportamiento se mantiene igual.

#### **Desventajas:**

Muy posible alta redundancia.

### **2. Cascada:**

#### **Ventajas:**

- ON DELETE CASCADE

Al eliminar una fila con una llave a la que hacen referencia las llaves foráneas de las filas existentes de otras tablas, las filas que dependan de esta llave foránea **también serán borradas**.

- ON UPDATE CASCADE

Caso análogo al de arriba, al actualizarse, las dependientes, también se actualizarán.

**Desventajas:**

Con un mal diseño de base de datos, se puede perder información, usando CASCADE no se puede especificar si una columna timestamp es parte de una llave foránea o de la clave con referencia.