***EJERCICIO 14***

*Objetivo del ejercicio:*

1. *Aprender el concepto de tipo de Entidad Supertipo. También llamada Superclase, Generalización o Especialización.*
2. *Aprender los tipos Entidad Supertipo.*
3. *Repasar Tipo de Correspondencia 1 a 0 en la que la FK es también PK.*

Se desea almacenar información de los empleados de una empresa, de todos ellos queremos almacenar el dni y el nombre. Los empleados pueden desempeñar el puesto de directivo, técnico o comercial. Para los empleados directivos debemos almacenar también el departamento. Para los técnicos, cuál es la máquina en la que están especializados. Y para los comerciales, cuál es la comisión que reciben.

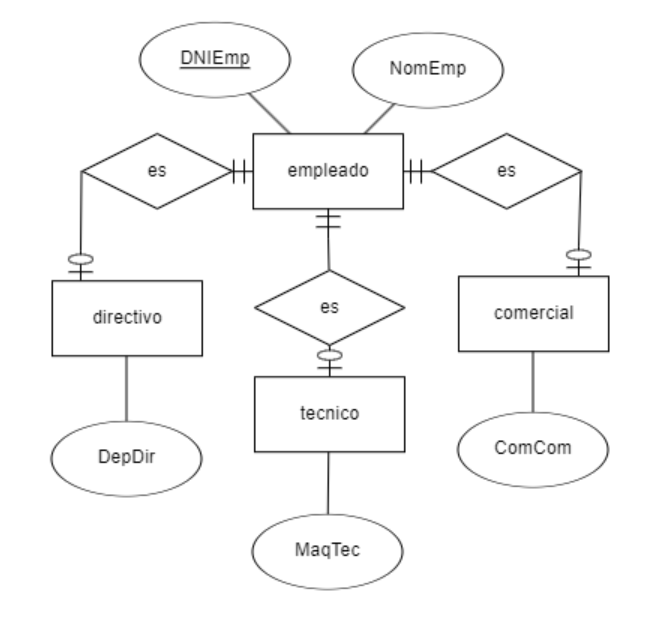
Se pide:

1. Modelar la base de datos. Para ello haremos:

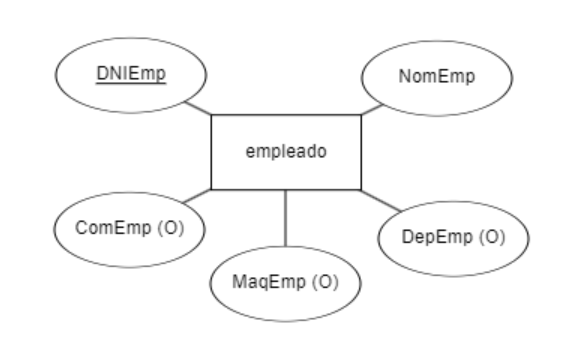
Nota: En este ejercicio, para poder explicaros mejor el concepto de Supertipo, quiero que almacenéis el departamento del directivo como un atributo que el cliente pueda introducir la cadena de caracteres que desee, igual con la maquina del técnico y con la comisión del comercial. Además, elegir como clave primaria el DNI del empleado.

* 1. Diseño Conceptual de Datos utilizando un Diagrama o Modelo Entidad- Relación. Lo hacemos en papel y lo pasamos a la Herramienta CASE ERD Plus.

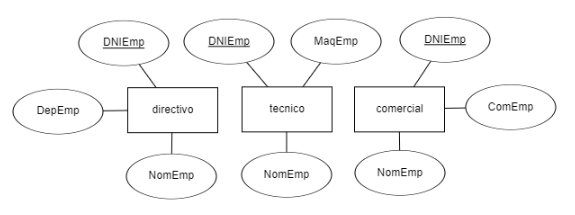
**Caso 1:**

****

**Caso 2:**

****

**Caso 3:**

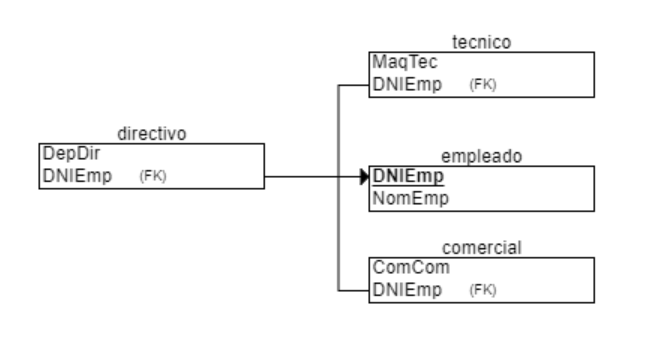


* 1. Diseño Lógico de Datos utilizando un Diagrama de Estructura de datos (DED). Lo hacemos en papel y lo pasamos a la Herramienta CASE MySql Workbench. En este apartado también vamos a poner el Diagrama Referencial que genera ERD Plus a partir del Modelo Entidad- Relación.

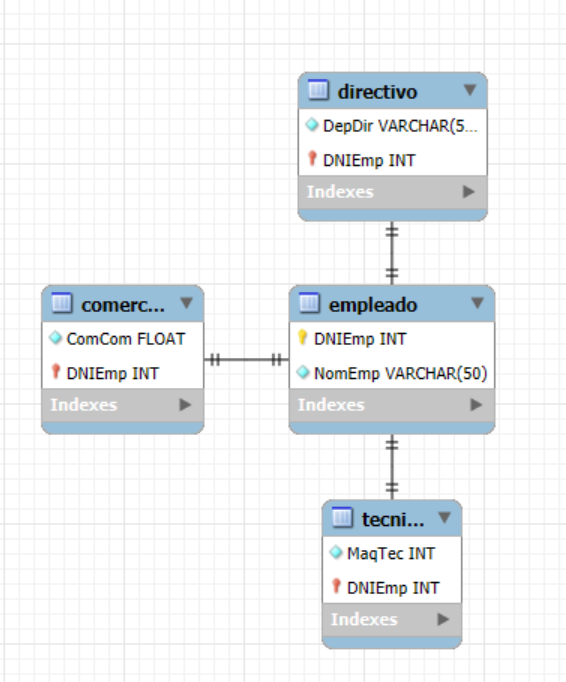
Recuerda que el Diseño Lógico de Datos es hacer el modelo relacional y para ello podemos hacer un DED o un Diagrama Referencial.

**Caso 1:**

**Diagrama Referencial:**

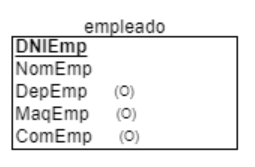
****

**DED:**

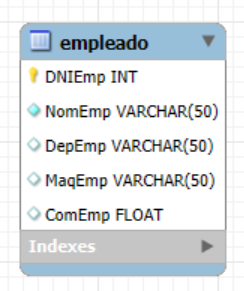
****

**Caso 2:**

**Diagrama Referencial:**

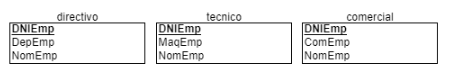
****

**DED:**

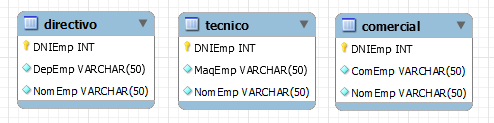
****

**Caso 3:**

**Diagrama Referencial:**

****

**DED:**



c. Diseño Físico de Datos. Creamos la base de datos y las tablas en SQL.

**Caso 1:**

CREATE DATABASE Ejercicio14\_Caso1;

CREATE TABLE empleado

(

DNIEmp INT NOT NULL,

NomEmp VARCHAR(50) NOT NULL,

PRIMARY KEY (DNIEmp)

);

CREATE TABLE directivo

(

DepDir VARCHAR(50) NOT NULL,

DNIEmp INT NOT NULL,

FOREIGN KEY (DNIEmp) REFERENCES empleado(DNIEmp)

);

CREATE TABLE tecnico

(

MaqTec VARCHAR(50) NOT NULL,

DNIEmp INT NOT NULL,

FOREIGN KEY (DNIEmp) REFERENCES empleado(DNIEmp)

);

CREATE TABLE comercial

(

ComCom FLOAT NOT NULL,

DNIEmp INT NOT NULL,

FOREIGN KEY (DNIEmp) REFERENCES empleado(DNIEmp)

);

**Caso 2:**

CREATE DATABASE Ejercicio14\_Caso2;

CREATE TABLE empleado

(

DNIEmp INT NOT NULL,

NomEmp VARCHAR(50) NOT NULL,

DepEmp VARCHAR(50),

MaqEmp VARCHAR(50),

ComEmp FLOAT,

PRIMARY KEY (DNIEmp)

);

**Caso 3:**

CREATE DATABASE Ejercicio14\_Caso3;

CREATE TABLE directivo

(

DNIEmp INT NOT NULL,

DepEmp VARCHAR(50) NOT NULL,

NomEmp VARCHAR(50) NOT NULL,

PRIMARY KEY (DNIEmp)

);

CREATE TABLE tecnico

(

DNIEmp INT NOT NULL,

MaqEmp VARCHAR(50) NOT NULL,

NomEmp VARCHAR(50) NOT NULL,

PRIMARY KEY (DNIEmp)

);

CREATE TABLE comercial

(

ComEmp FLOAT NOT NULL,

DNIEmp INT NOT NULL,

NomEmp VARCHAR(50) NOT NULL,

PRIMARY KEY (DNIEmp)

);

Insertar los mismos datos desde phpmyadmin en todas las tablas.

**Caso 1:**

INSERT INTO empleado (DNIEmp, NomEmp)

VALUES

(12345678, 'Juan Pérez'),

(23456789, 'Ana Gómez'),

(34567890, 'Carlos López'),

(45678901, 'Laura Martínez');

INSERT INTO directivo (DepDir, DNIEmp)

VALUES

('Recursos Humanos', 12345678),

('Finanzas', 23456789);

INSERT INTO tecnico (MaqTec, DNIEmp)

VALUES

('Servidor', 34567890),

('Impresora', 45678901);

INSERT INTO comercial (ComCom, DNIEmp)

VALUES

(2000.00, 23456789),

(1500.00, 34567890);

**Caso 2:**

INSERT INTO empleado (DNIEmp, NomEmp, DepEmp, MaqEmp, ComEmp) VALUES

(12345678, 'Juan Pérez', 'Recursos Humanos', NULL, NULL), (23456789, 'Ana Gómez', 'Finanzas', NULL, NULL), (34567890, 'Carlos López', NULL, 'Servidor', 1500.00), (45678901, 'Laura Martínez', NULL, 'Impresora', 1800.00);

**Caso 3:**

INSERT INTO directivo (DNIEmp, DepEmp, NomEmp)

VALUES

(12345678, 'Recursos Humanos', 'Juan Pérez'),

(23456789, 'Finanzas', 'Ana Gómez');

INSERT INTO tecnico (DNIEmp, MaqEmp, NomEmp)

VALUES

(34567890, 'Servidor', 'Carlos López'),

(45678901, 'Impresora', 'Laura Martínez');

INSERT INTO comercial (ComEmp, DNIEmp, NomEmp)

VALUES

(2000.00, 56789012, 'Marta Ruiz'),

(1500.00, 67890123, 'Luis Fernández');

**RESPONDE A LA SIGUIENTE PREGUNTA:**

**¿Cómo crees que se podría controlar lo siguiente?**

* Que un empleado solo puede ser directivo, o técnico o comercial. Es decir, solo puede tener un único cargo como empleado.

Usar una tabla central de control para gestionar los DNIs, implementar triggers que validen la unicidad del DNI al insertar datos en cualquiera de las tablas y validar mediante lógica en la aplicación externa.

* Que al dar de alta a un empleado, siempre especifiquemos:
  + Si es directivo, su departamento.
  + Si es técnico, la máquina.
  + Si es comercial, su comisión.

Definir correctamente los campos NOT NULL en las tablas, utilizar triggers para validar la inserción y complementar con validaciones en la aplicación externa o con procedimientos almacenados.