

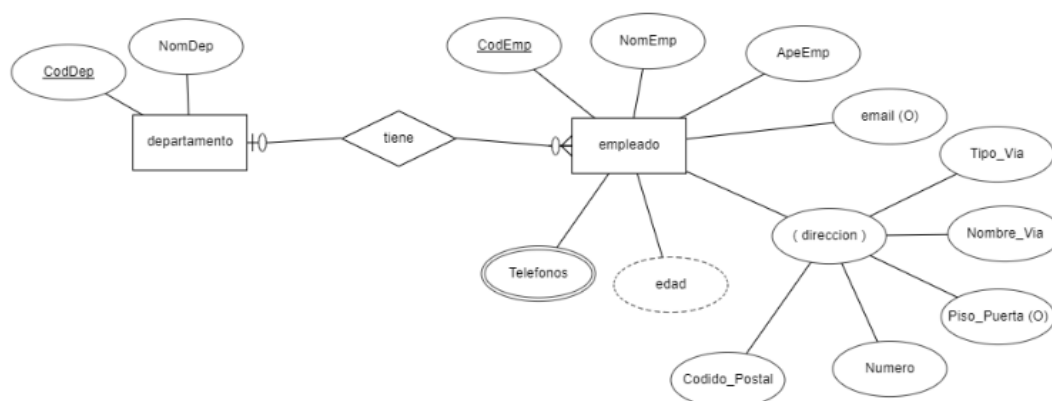
Objetivo del ejercicio:

1. Aprender el concepto de atributo multivaluado.
2. Repasar todos los tipos de atributos.

Se desea diseñar una base de datos para almacenar la información de los departamentos y de los empleados de una empresa. De los departamentos se desea almacenar el código y el nombre. De los empleados almacenamos su código, nombre y apellidos, la dirección completa, el email (que no es obligatorio), la edad y los teléfonos. Un empleado puede no tener departamento momentáneamente. Además es posible tener departamentos de reciente creación que aún no tengan empleados.

Se pide:

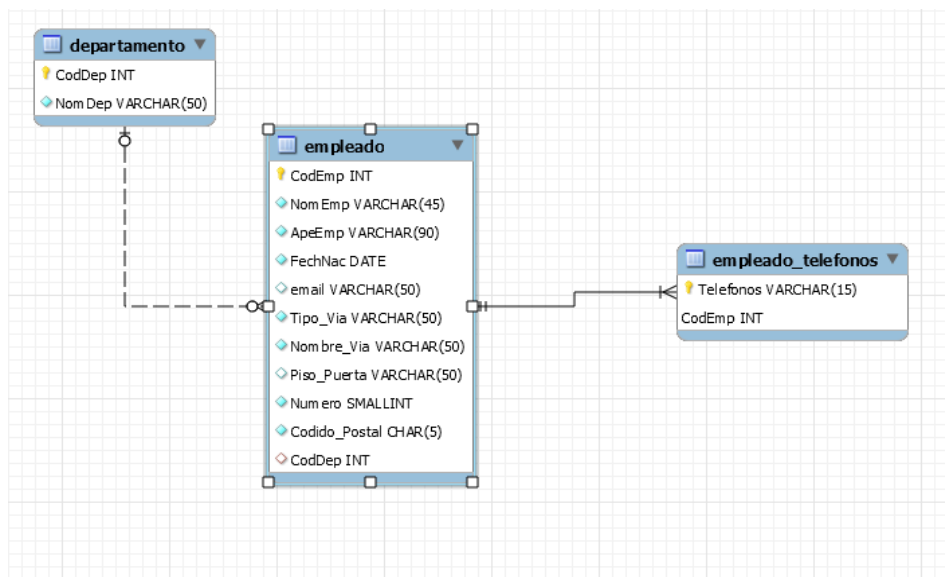
1. Modelar la base de datos. Para ello haremos:
 - Diseño Conceptual de Datos utilizando un Diagrama o Modelo Entidad-Relación. Lo hacemos en papel y lo pasamos a la Herramienta CASE ERD Plus.
(Importar el fichero Ejercicio 18 ER.erdplus)



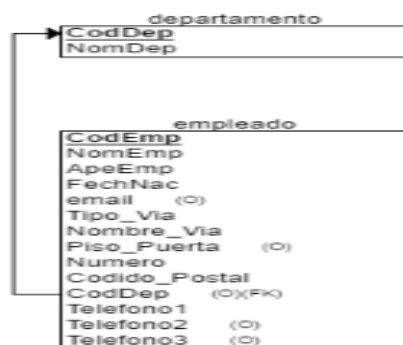
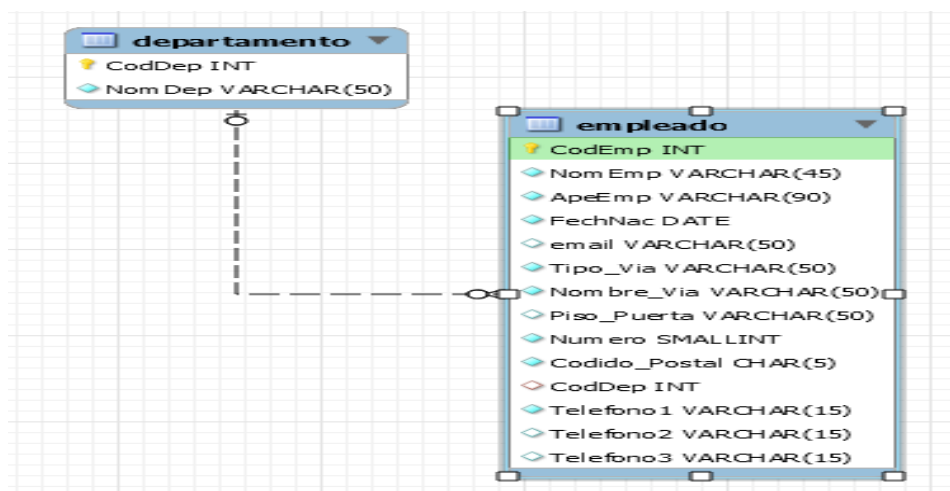
- b. Diseño Lógico de Datos utilizando un Diagrama de Estructura de datos (DED). Lo hacemos en papel y lo pasamos a la Herramienta CASE MySQL Workbench. En este apartado también vamos a poner el Diagrama Referencial que genera ERD Plus a partir del Modelo Entidad-Relación. Recuerda que el Diseño Lógico de Datos es hacer el modelo relacional y para ello podemos hacer un DED o un Diagrama Referencial.

Caso 1: No sabemos cuántos teléfonos puede tener el empleado y además nuestro cliente quiere almacenar todos los que el empleado le proporcione. Pero al menos debe proporcionarle uno. Así es como lo modela por defecto ERD Plus. Observa que lo que hace es crear una entidad débil con dependencia en identificación.

Diagrama Referencial

DED

Caso 2: Nuestro cliente nos ha dicho que como mucho quiere almacenar tres teléfonos de cada empleado siendo solo uno obligatorio.

Diagrama ReferencialDED

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

Caso1

```
CREATE DATABASE EJERCICIO18_CASO1;  
USE EJERCICIO18_CASO1;
```

```
CREATE TABLE departamento  
(  
    CodDep INT NOT NULL,  
    NomDep varchar(50) NOT NULL,  
    PRIMARY KEY (CodDep)  
);
```

```
CREATE TABLE empleado  
(  
    CodEmp INT NOT NULL,  
    NomEmp varchar(45) NOT NULL,  
    ApeEmp varchar(90) NOT NULL,  
    FechNac DATE NOT NULL,  
    email varchar(50),  
    Tipo_Via varchar(50) NOT NULL,  
    Nombre_Via varchar(50) NOT NULL,  
    Piso_Puerta varchar(50),  
    Numero SMALLINT NOT NULL,  
    Codido_Postal CHAR(5) NOT NULL,  
    CodDep INT,  
    PRIMARY KEY (CodEmp),  
    FOREIGN KEY (CodDep) REFERENCES departamento(CodDep)  
);
```

```
CREATE TABLE empleado_Telefonos  
(  
    Telefonos VARCHAR(15) NOT NULL,  
    CodEmp INT NOT NULL,  
    PRIMARY KEY (Telefonos, CodEmp),  
    FOREIGN KEY (CodEmp) REFERENCES empleado(CodEmp) ON UPDATE NO ACTION ON DELETE  
    CASCADE  
);
```

Caso2

```
CREATE DATABASE EJERCICIO18_CASO2;  
USE EJERCICIO18_CASO2;
```

```
CREATE TABLE departamento  
(  
    CodDep INT NOT NULL,  
    NomDep varchar(50) NOT NULL,  
    PRIMARY KEY (CodDep)  
);
```

```

CREATE TABLE empleado
(
  CodEmp INT NOT NULL,
  NomEmp varchar(45) NOT NULL,
  ApeEmp varchar(90) NOT NULL,
  FechNac DATE NOT NULL,
  email varchar(50),
  Tipo_Via varchar(50) NOT NULL,
  Nombre_Via varchar(50) NOT NULL,
  Piso_Puerta varchar(50),
  Numero SMALLINT NOT NULL,
  Codido_Postal CHAR(5) NOT NULL,
  Telefono1 VARCHAR(15) NOT NULL,
  Telefono2 VARCHAR(15),
  Telefono3 VARCHAR(15),
  CodDep INT,
  PRIMARY KEY (CodEmp),
  FOREIGN KEY (CodDep) REFERENCES departamento(CodDep) ON UPDATE NO ACTION ON
  DELETE SET NULL
);

```

2. Insertar tres empleados (el primero con tres teléfonos y el segundo con un teléfono) desde phpmyadmin utilizando la sentencia INSERT INTO del LMD de SQL.

Caso 1

```

INSERT INTO departamento (CodDep,NomDep)
VALUES (1,'Dep1'),(2,'Dep2');

```

```

INSERT INTO empleado(CodEmp, NomEmp, ApeEmp, FechNac,email, Tipo_Via, Nombre_Via,
Piso_Puerta, Numero, Codido_Postal, CodDep)
VALUES (1, 'José', 'Pérez Pérez', '1990/03/03','josepp@gmail.com','Calle','Calle1',NULL,1,'29590',1),
(2, 'Ana', 'García García', '1998/04/04' , 'anagg@gmail.com','Calle','Calle2',NULL,2,'29590',1);

```

```
INSERT INTO empleado_telefonos(Telefonos, CodEmp)
VALUES ('666123456',1),
      ('666123457',1),
      ('666123458',1),
      ('606123456',2);
```

Caso 2

```
INSERT INTO departamento (CodDep,NomDep)
VALUES (1,'Dep1'),(2,'Dep2');
```

```
INSERT INTO empleado(CodEmp, NomEmp, ApeEmp, FechNac, email, Tipo_Via, Nombre_Via,
Piso_Puerta, Numero, Codido_Postal, Telefono1, Telefono2, Telefono3, CodDep)
VALUES
(1, 'José', 'Pérez Pérez', '1990/03/03', 'josepp@gmail.com','Calle','Calle1',NULL,1,'29590','666123456',
'666123457', '666123458',1),
(2, 'Ana', 'García García', '1998/04/04', 'anagg@gmail.com','Calle','Calle2',NULL,2,'29590',
'606123456',NULL,NULL,1);
```

3. Realizar la siguientes consulta:

- Obtener todos los teléfonos del empleado José Pérez Pérez para ambos casos de diseños.

```
/*Caso 1*/
```

```
SELECT telefonos
FROM empleado e JOIN empleado_telefonos t ON (e.CodEmp=t.CodEmp)
WHERE NomEmp='José' AND ApeEmp='Pérez Pérez'
```

```
/*Caso 2*/
```

```
SELECT CONCAT(Telefono1, ' ',IFNULL(Telefono2,' '), ' ', IFNULL(Telefono3,' ')) AS Teléfonos
FROM empleado
WHERE NomEmp='José' AND ApeEmp='Pérez Pérez'
```