



## **MÁSTER BIG DATA Y DATA SCIENCE**

### **CREACIÓN DE UNA BASE DE DATOS**

*Análisis, diseño e implementación de una base de datos*

**MÓDULO:** Structured Query Language (SQL)

**PROFESORA:** Isabel Riomoros

**AUTOR:** Hassan Chafi Xavier

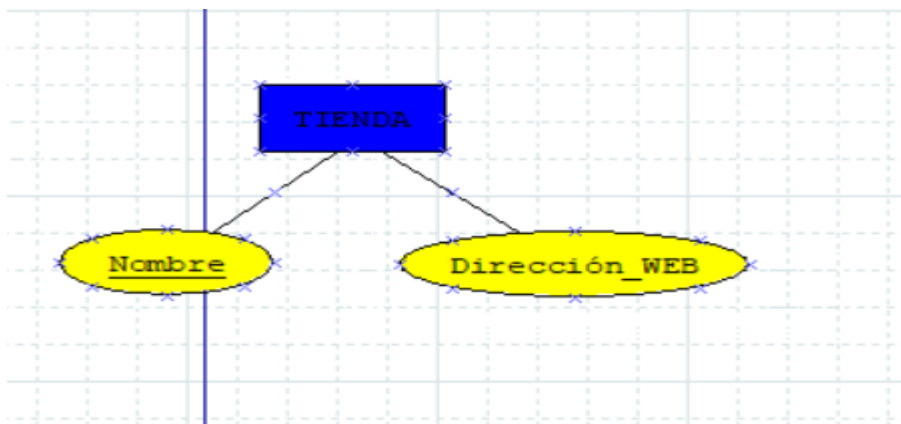
## DISEÑO Y ANALISIS DEL MODELO ENTIDAD – RELACION

“De la tienda sabemos el nombre, que es distinto para cada tienda, quien la gestiona (Android, Apple, Amazon, ...) y dirección web.”

La primera entidad que se identifica es la tienda de aplicaciones que es un objeto abstracto ya que la tienda de aplicaciones no existe físicamente como tal y la cual tiene dos atributos que señalan las propiedades de la tienda de aplicación:

- Nombre → Es la clave primaria de la entidad que determina las instancias de manera única. Se trata de un atributo simple que no puede descomponerse en otros atributos, univalorado ya que sólo toma un valor de forma simultánea y cuyo dominio es el conjunto de los nombres existentes de tiendas de aplicaciones.
- Dirección WEB → Se trata de un atributo simple que no admite descomposición en otros atributos, univalorado ya que admite un único valor de manera simultánea y cuyo dominio se restringe a la dirección web asociada a la tienda de aplicaciones definida en el atributo anterior.

**Imagen 1. Entidad tienda y sus atributos.**



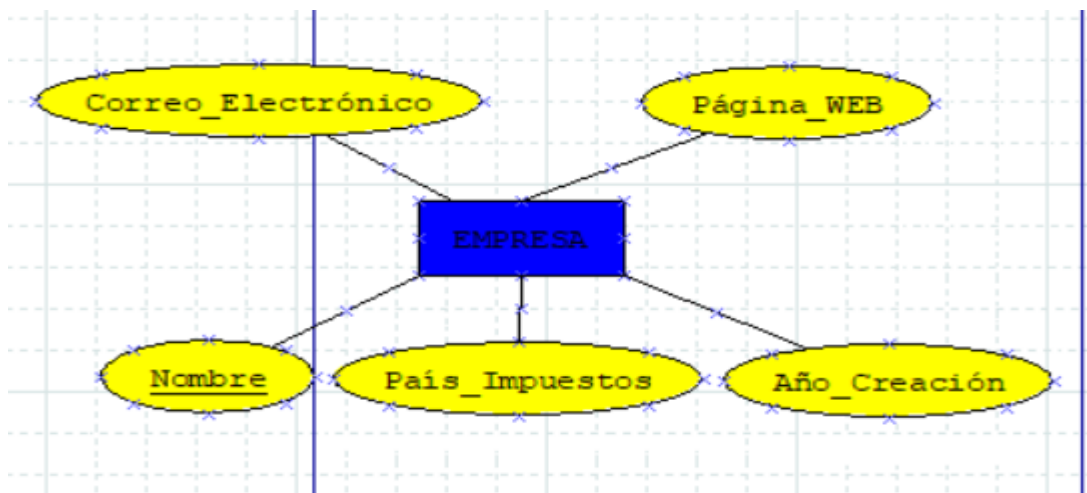
“De las empresas de servicios que realizan las aplicaciones (apps), conocemos su nombre, país en el que paga sus impuestos, año de creación, correo electrónico y pagina web, sabemos que estas empresas no tienen ninguna conexión con las tiendas de aplicaciones.”

La siguiente entidad que se identifica es la empresa que realiza las aplicaciones que es un objeto concreto ya que es un lugar físico que reúne personas encargadas del desarrollo de estas aplicaciones y que tiene cinco atributos:

- Nombre → Es la clave primaria de la entidad que determina las instancias de manera única. Se trata de un atributo simple que no se descompone en otros atributos, univalorado ya que admite un único valor de forma simultánea y cuyo dominio es el conjunto de los nombres de las empresas fabricantes de aplicaciones.

- País de impuestos → Se trata de un atributo simple que no se descompone en más atributos, univalorado ya que sólo se pagan impuestos en un país de manera simultánea y cuyo dominio es el conjunto de países que existen en el mundo que adoptará su valor dependiendo de cual sea el país en el que paga impuestos la empresa para la que se está realizando su registro.
- Año de creación → Se trata de un atributo simple que no se descompone en otros atributos, univalorado puesto que la empresa sólo ha podido ser creada en un año concreto y cuyo dominio es el conjunto de los años desde que se comenzaron a crear empresas desarrolladoras de aplicaciones hasta la actualidad y que adoptará su valor en función de cual sea el año en el que se creó la empresa para la cual se está realizando el registro.
- Correo electrónico → Se trata de un atributo simple que no se descompone en más atributos, univalorado ya que estas empresas sólo disponen de un correo electrónico y cuyo dominio es el correo electrónico de la empresa para la cual se está realizando el registro.
- Página WEB → Se trata de un atributo simple que no se descompone en otros atributos, univalorado ya que la empresa sólo dispone de una página web que la representa y cuyo dominio es la página web de la empresa para la cual se está realizando el registro.

**Imagen 2. Entidad empresa y sus atributos.**



“En la empresa hay empleados, pero debido al dinamismo en estos tipos de trabajo y, a la oferta y la demanda en el sector, el empleado puede haber trabajado en varias empresas del sector e incluso puede trabajar en la misma empresa en distintos periodos de tiempo, nos interesa conocer la experiencia profesional del empleado. Además, del empleado nos interesa el DNI, dirección (calle, número, código postal), correo electrónico y teléfono.”

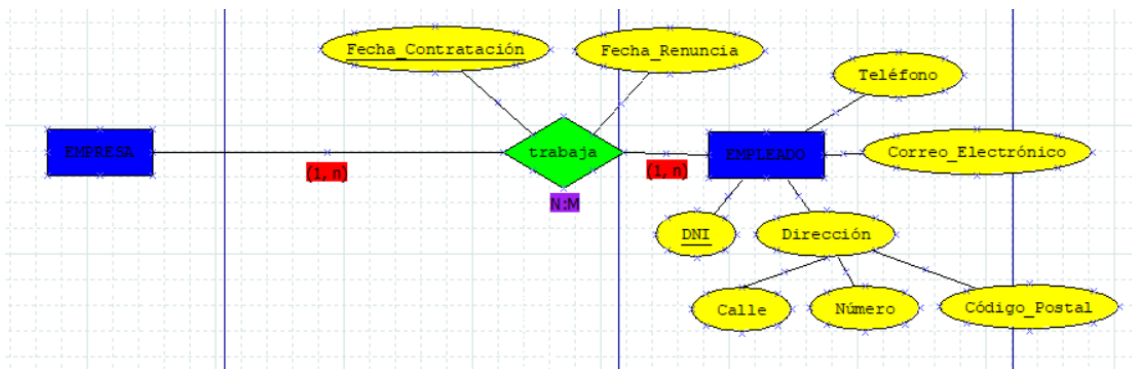
La siguiente entidad identificada es el empleado de la empresa que es un objeto concreto ya que físicamente es tangible y para el que se identifican directamente cuatro atributos:

- DNI → Es la clave primaria de la entidad que determina las instancias de manera única. Se trata de un atributo simple que no se descompone en otros atributos, univalorado ya que sólo admite un valor de manera simultánea y cuyo dominio es el conjunto de documentos identificativos.
- Dirección → Se trata de un atributo univalorado ya que en cada registro sólo se asocia una dirección de manera simultánea, pero compuesto por los siguientes atributos:
  - Calle → Se trata de un atributo simple que no se descompone en más atributos, univalorado ya que sólo admite un valor simultáneamente y cuyo dominio es el nombre de la calle en la que reside el empleado.
  - Número → Se trata de un atributo simple que no se descompone en más atributos, univalorado ya que sólo admite un valor simultáneamente y cuyo dominio es el número de la calle en la que reside el empleado.
  - Código Postal → Se trata de un atributo simple que no se descompone en más atributos, univalorado ya que sólo admite un valor simultáneamente y cuyo dominio es el código postal asociado a la calle en la que reside el empleado.
- Correo electrónico → Se trata de un atributo simple que no está compuesto por otros atributos, univalorado ya que cada empleado tiene sólo un correo electrónico y cuyo dominio es el correo electrónico del empleado para el cual se realiza el registro.
- Teléfono → Se trata de un atributo simple que no se descompone en otros atributos, univalorado ya que cada empleado dispone de un sólo número de teléfono simultáneamente y cuyo dominio es el número de teléfono del empleado para el cual se realiza el registro.

De la entidad empleado se identifica una relación de muchos a muchos con la entidad empresa. Un empleado puede trabajar en varias empresas (debido al dinamismo del sector, el empleado puede haber trabajado en otras empresas del sector o en la misma empresa, pero en periodos diferentes) y una empresa puede tener varios empleados. De esta relación nacen los siguientes atributos:

- Fecha de contratación → Es la clave primaria de la entidad que determina las instancias de manera única. Se trata de un atributo simple que no se descompone en más atributos, univalorado ya que solo admite un valor de forma simultánea y cuyo dominio se restringe a la fecha en la que se contrató al empleado para el cual se realiza el registro.
- Fecha de renuncia → Se trata de un atributo simple que no se descompone en otros atributos, univalorado ya que solo admite un valor de manera simultánea y cuyo dominio se restringe a la fecha en la que el empleado dejó de formar parte de la empresa (en caso de estar trabajando en la empresa en la actualidad, aparecerá como defecto la fecha actual).

**Imagen 3. Entidad empleado, sus atributos y su relación con la entidad empresa.**



“En la empresa se realizan aplicaciones, de las que conocemos su nombre que es único, el código de aplicación, la fecha en la que se comenzó a realizar y fecha de terminación, la categoría o categorías en las que se puede incluir (entretenimiento, social, educación, ...), espacio de memoria, precio.”

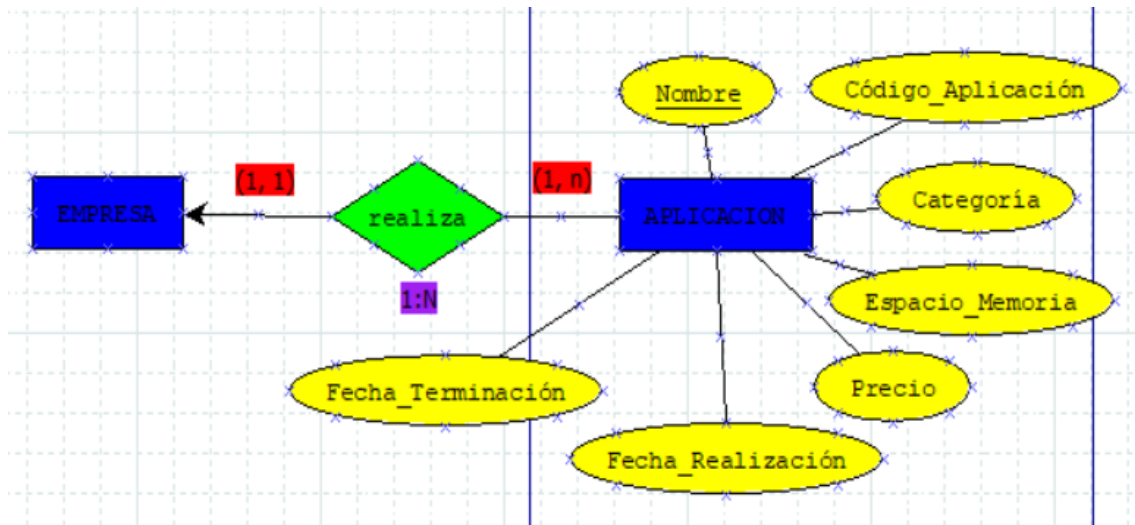
La siguiente entidad que se identifica es la de las aplicaciones que son un objeto abstracto de naturaleza intangible y que tiene siete atributos:

- Nombre → Es la clave primaria de la entidad que determina las instancias de manera única. Se trata de un atributo simple que no se descompone en otros atributos, univalorado ya que sólo admite un valor de forma simultánea y cuyo dominio es el conjunto de nombres de aplicaciones existentes.
- Código de la aplicación → Se trata de un atributo simple que no se descompone en más atributos, univalorado puesto que solo tiene un valor de manera simultánea y cuyo dominio se restringe al código de la aplicación para la cual se está realizando el registro.
- Categoría → Se trata de un atributo simple que no se descompone en otros atributos, univalorado ya que sólo admite un valor simultáneamente y cuyo dominio se encuentra dentro de una lista de posibles categorías dadas (entretenimiento, social, educación...).
- Espacio de memoria → Se trata de un atributo simple que no se descompone en otros atributos, univalorado ya que solo tiene un valor simultáneamente y cuyo dominio es el espacio de memoria que tiene la aplicación para la cual se realiza el registro.
- Precio → Se trata de un atributo simple que no se descompone en otros atributos, univalorado ya que solo tiene un valor de manera simultánea y cuyo dominio es el precio que lleva asociado la aplicación para la cual se está realizando el registro.
- Fecha de realización → Se trata de un atributo simple que no se descompone en más atributos, univalorado ya que sólo admite un valor simultáneamente y cuyo dominio es la fecha en la que se comenzó a desarrollar la aplicación.

- Fecha de terminación → Se trata de un atributo simple que no se descompone en más atributos, univalorado ya que sólo admite un valor simultáneamente y cuyo dominio es la fecha en la que se terminó de desarrollar la aplicación

Existe una relación de uno a muchos entre la entidad aplicación y la entidad empresa. Una aplicación es realizada por una empresa, pero una empresa puede realizar varias aplicaciones.

**Imagen 4. Entidad aplicación, sus atributos y su relación con la entidad empresa.**



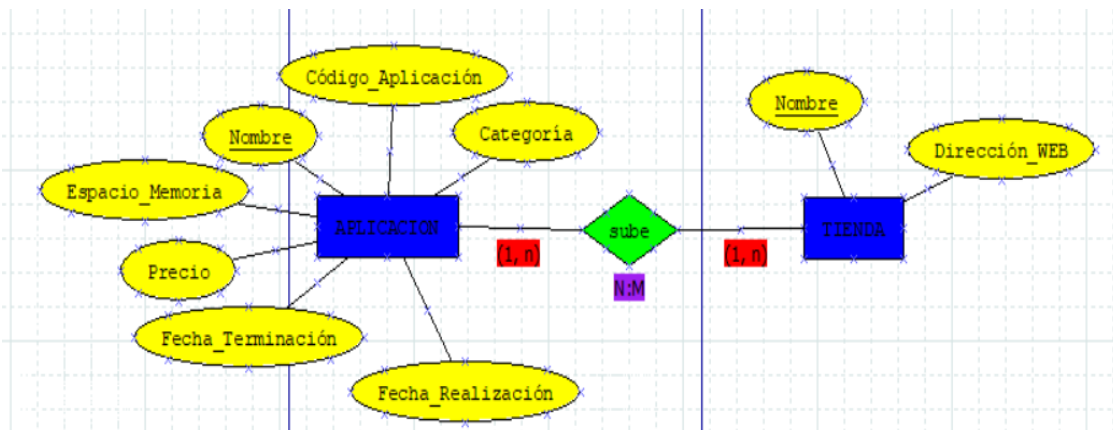
“Las aplicaciones son subidas a las tiendas o plataformas. Una misma aplicación puede ser subida a varias tiendas, por supuesto una tienda tiene muchas aplicaciones.”

A continuación, se identifica una relación de muchos a muchos entre la entidad aplicación y la entidad tienda. Una aplicación puede ser subida a varias tiendas y una tienda puede disponer de varias aplicaciones.

De esta relación surgen los siguientes atributos:

- Nombre de Aplicación → Es una clave ajena y se refiere a la clave primaria definida en la entidad aplicación.
- Nombre de Tienda → Es una clave ajena y se refiere a la clave primaria definida para la entidad tienda.

**Imagen 5. Entidad aplicación y su relación con la entidad tienda.**



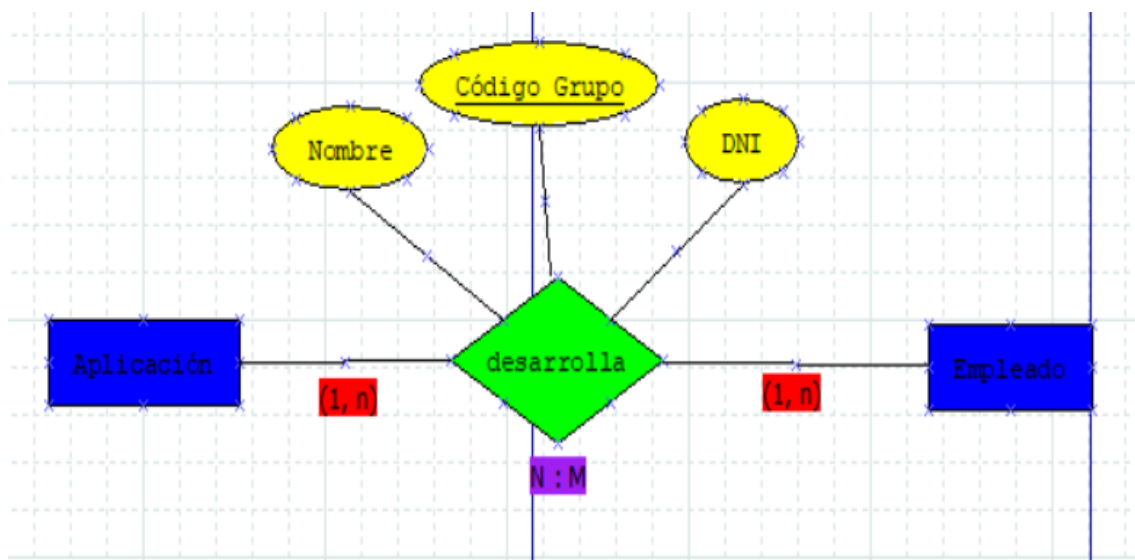
“Cada aplicación está realizada por un grupo de empleados.”

Se identifica una relación de muchos a muchos entre la entidad aplicación y la entidad empleado. Un grupo de empleados pueden desarrollar varias aplicaciones y una aplicación está desarrollada por varios empleados.

De esta relación surgen los siguientes atributos:

- Nombre → Es una clave ajena y se refiere al atributo nombre definido para la entidad aplicación.
- DNI → Es una clave ajena y se refiere al atributo DNI definido para la entidad empleado.
- Código de Grupo → Es la clave primaria de la entidad que determina las instancias de manera única. Se trata de un atributo simple que no se descompone en otros atributos, univalorado ya que sólo admite un valor de forma simultánea y cuyo dominio es el conjunto de grupos desarrolladores de aplicaciones.

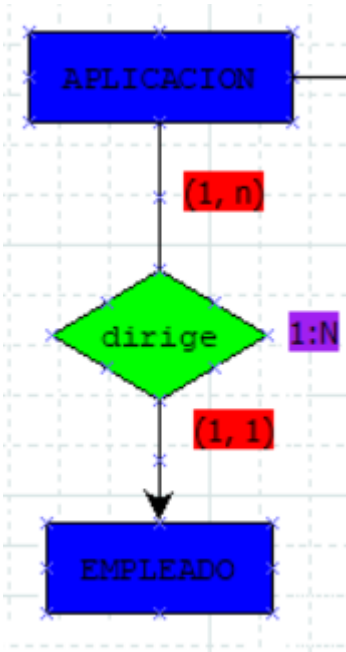
**Imagen 6. Relación entre la entidad aplicación y la entidad empleado.**



“La aplicación es dirigida por un empleado y un empleado puede dirigir varias aplicaciones.”

La siguiente relación es de muchos a uno y se produce entre la entidad aplicación y la entidad de empleado. Un empleado puede dirigir varias aplicaciones, pero una aplicación está dirigida sólo por un empleado.

**Imagen 7. Entidad aplicación y su relación con la entidad empleado.**



“Un usuario puede descargar o no aplicaciones, pero no puede descargar dos veces la misma aplicación. El usuario puede puntuar de 0 a 5 cada una de las aplicaciones que se descarga y hacer comentarios referentes a la misma. Del usuario conocemos el número de cuenta que es único, nombre, dirección y si se descarga la aplicación en el teléfono conocemos el número de móvil.”

Se identifica la entidad usuarios que es un objeto concreto que son quienes se descargan las aplicaciones y que tiene 4 atributos:

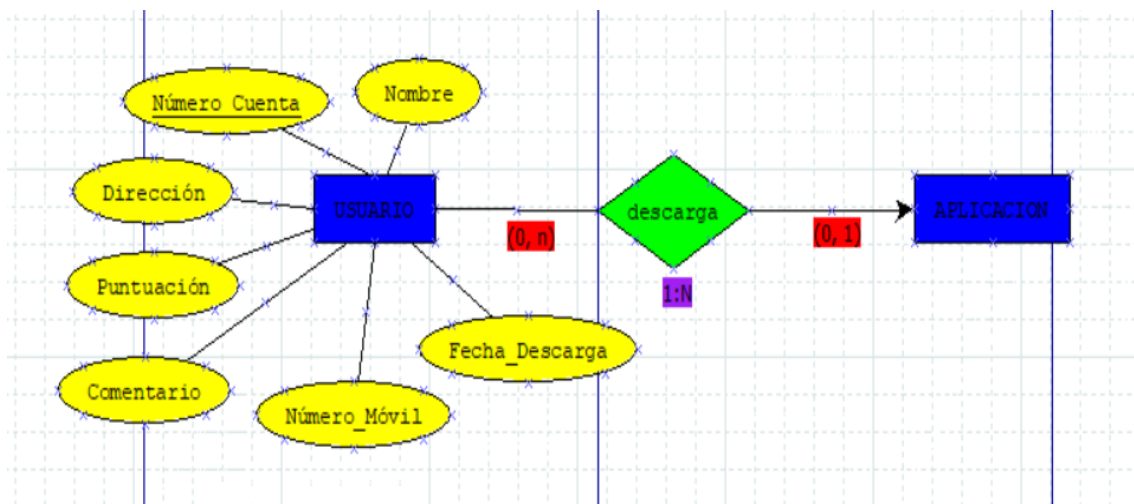
- Número de cuenta → Es la clave primaria de la entidad que determina las instancias de manera única. Se trata de un atributo simple que no admite descomposición en otros atributos, univalorado ya que sólo tiene un valor simultáneamente y cuyo dominio es el conjunto de números de cuenta.
- Nombre → Se trata de un atributo simple que no se descompone en otros atributos, univalorado ya que sólo tiene un valor simultáneamente y cuyo dominio es el nombre asociado al número de cuenta para el cual se realiza el registro.
- Dirección → Se trata de un atributo simple que no se descompone en otros atributos, univalorado ya que sólo admite un valor simultáneamente y cuyo dominio es la dirección asociada al nombre del usuario para el que se realiza el registro.



- Puntuación → Se trata de un atributo simple que no se descompone en otros atributos, univalorado ya que sólo admite un valor simultáneamente y cuyo dominio se encuentra entre 0 y 5 dependiendo de la elección del usuario al valorar su satisfacción con la aplicación.
- Comentario → Se trata de un atributo simple que no se descompone en otros atributos, univalorado ya que sólo recibe un único valor de manera simultánea y cuyo dominio es una cadena de caracteres que el usuario decide escribir en función de su opinión acerca de la aplicación.
- Fecha de descarga → Se trata de un atributo simple que no se descompone en otros atributos, univalorado ya que sólo recibe una fecha de tiempo simultáneamente y cuyo dominio es la fecha efectiva en la que el usuario se ha descargado una determinada aplicación.
- Número de móvil → Se trata de un atributo simple que no se descompone en otros atributos, univalorado puesto que cada usuario sólo tiene un número de móvil desde el que realiza la descarga de la aplicación y cuyo dominio se restringe al número de móvil desde el que se realiza la descarga de la aplicación.

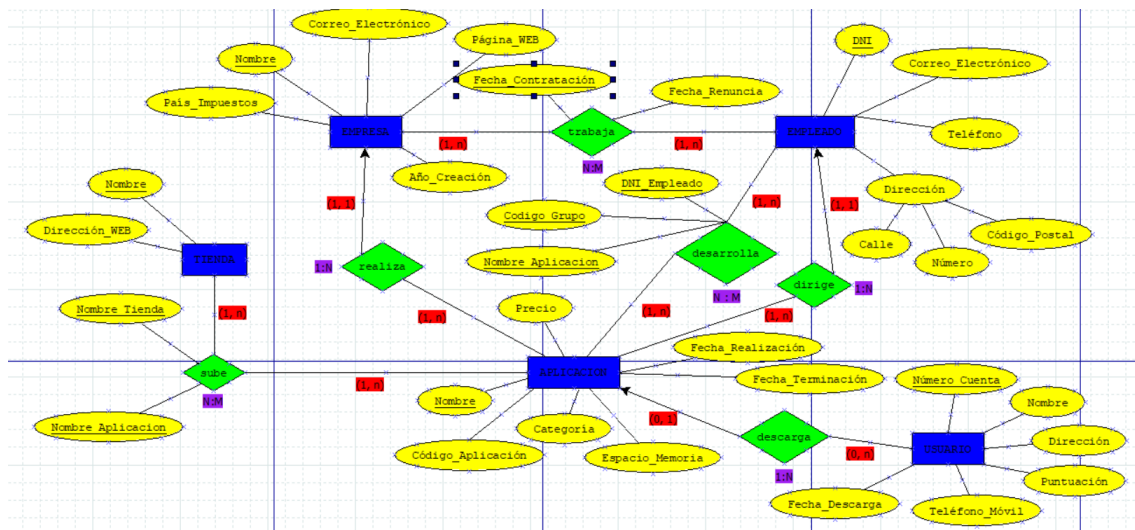
Se identifica una relación de uno a muchos entre la entidad usuario y la entidad aplicación. Un usuario puede descargar una vez la aplicación (o puede decidir no descargarse la aplicación nunca) y una aplicación puede ser descargada por varios usuarios simultáneamente (o puede no ser descargada por nadie).

**Imagen 8. Entidad usuario, sus atributos y su relación con la entidad aplicación.**



Una vez que ya se han definido las entidades y los atributos que revelan sus propiedades, así como las características de estos atributos y las relaciones que surgen entre las distintas entidades; el modelo entidad-relación queda diseñado tal y como se puede apreciar en la imagen 9 que se visualiza a continuación.

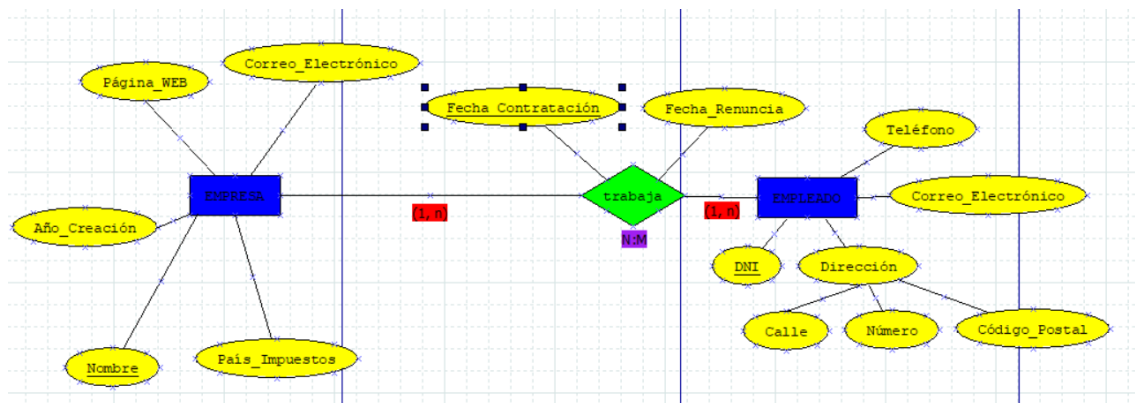
**Imagen 9. Modelo Entidad – Relación.**



## TRANSFORMACION DEL MODELO ENTIDAD – RELACIÓN EN UN MODELO RELACIONAL

El primer paso a tablas que se realiza es el que se corresponde con la entidad empresa y la entidad empleado y su relación. Al tratarse de una relación de muchos a muchos, la relación entre estas dos entidades genera una nueva tabla con las claves primarias de ambas entidades que relaciona.

**Imagen 10. Entidad empresa, entidad empleado, sus atributos y su relación.**



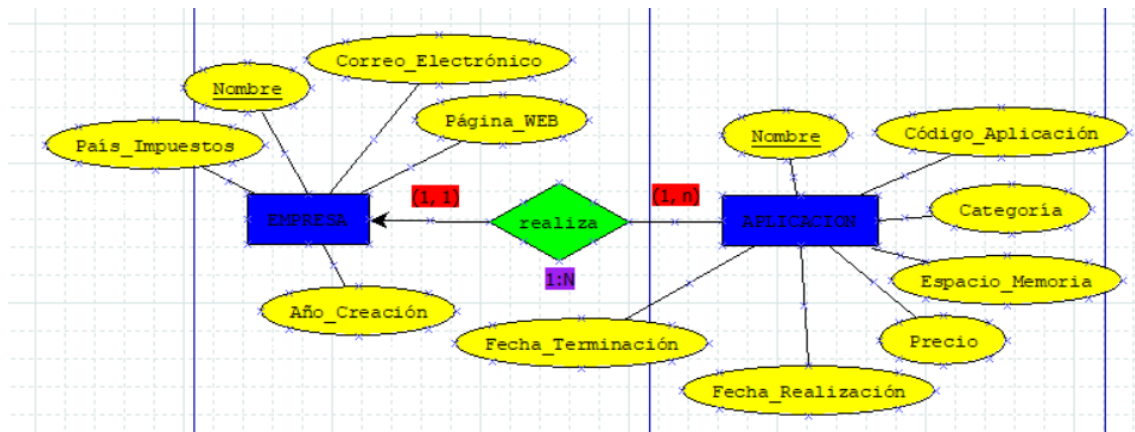
EMPRESA (Nombre, País\_Impuestos, Correo\_Electrónico, Año\_Creación, Página\_WEB)

EMPLEADO (DNI, Dirección\_Calle, Dirección\_Número, Dirección\_Código\_Postal, Correo\_Electrónico, Teléfono)

TRABAJA (Nombre\_Empresa, DNI, Fecha\_Contratación, Fecha\_Renuncia)

La siguiente relación es la que se corresponde con la entidad de empresa y la entidad de aplicación. Se trata de una relación de uno a muchos por lo que la clave primaria del lado de uno se añade a la tabla del lado de varios.

**Imagen 10. Entidad empresa, entidad aplicación, sus atributos y su relación.**

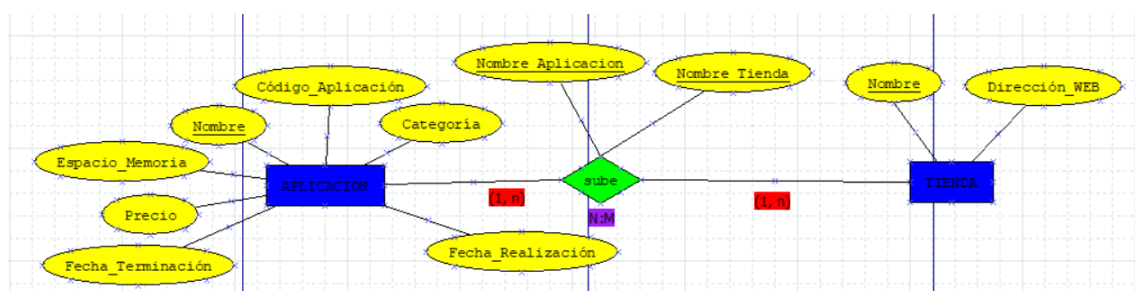


EMPRESA (Nombre, País\_Impuestos, Correo\_Electrónico, Año\_Creación, Página\_WEB)

APLICACIÓN (Nombre, Código\_Aplicación, Categoría, Espacio\_Memoria, Precio, Fecha\_Realización, Fecha\_Terminación, Nombre Empresa)

La siguiente relación se trata entre la entidad aplicación y la entidad tienda. Al tratarse de una relación de muchos a muchos, se genera una nueva tabla entre las dos entidades resultante de la relación que las une.

**Imagen 11. Entidad aplicación, entidad tienda, sus atributos y su relación.**



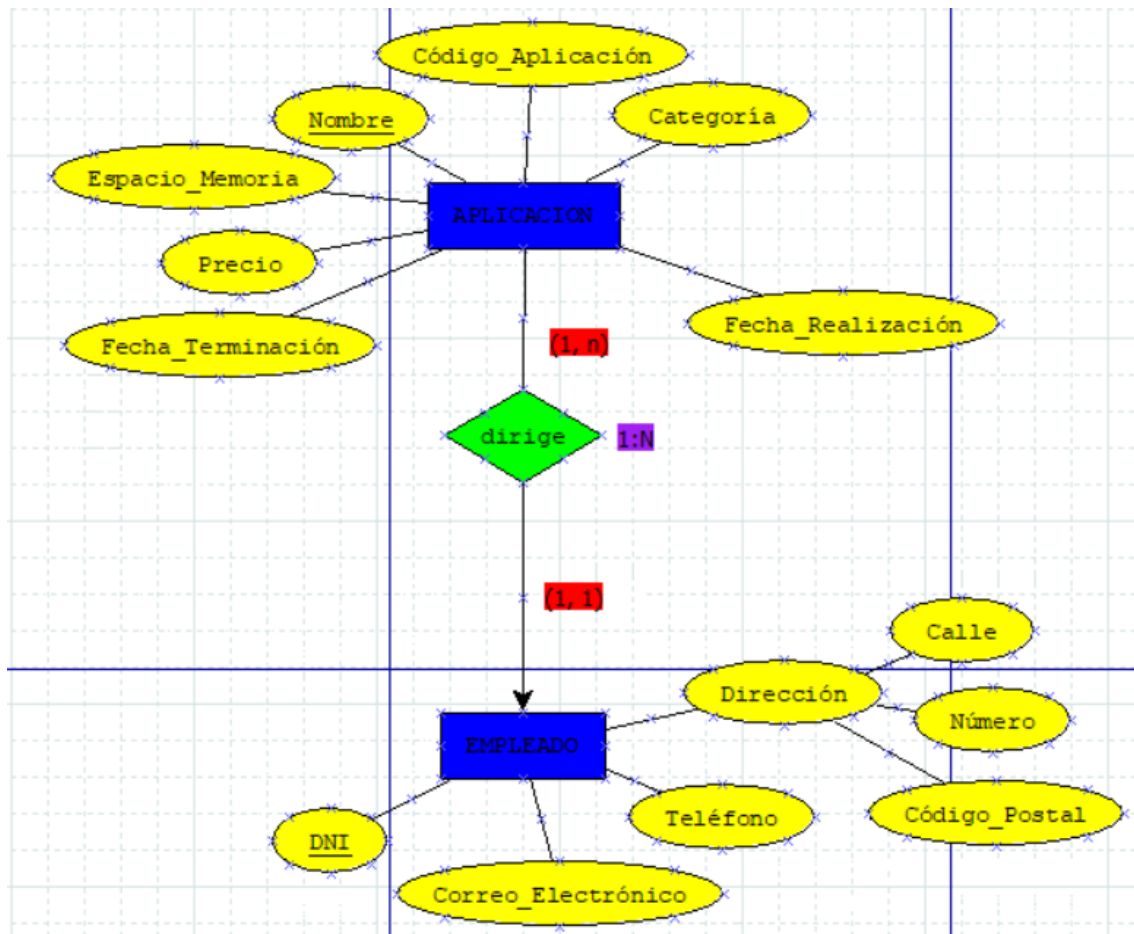
APLICACIÓN (Nombre, Código\_Aplicación, Categoría, Espacio\_Memoria, Precio, Fecha\_Realización, Fecha\_Terminación)

TIENDA (Nombre, Dirección\_WEB)

SUBE (Nombre Aplicación, Nombre Tienda)

La relación encontrada entre la entidad aplicación y la entidad empleado es de uno a muchos. Por lo tanto, el paso a tablas consistirá en poner la clave primaria del lado uno en el lado de varios.

**Imagen 12. Entidad aplicación, entidad empleado, sus atributos y su relación.**

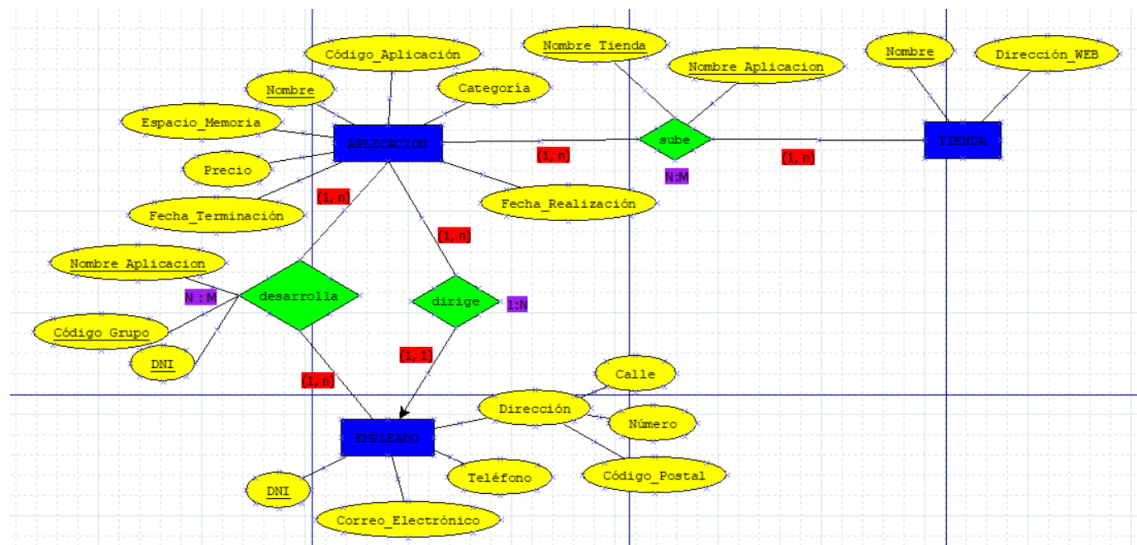


APLICACIÓN (Nombre, Código\_Aplicación, Categoría, Espacio\_Memoria, Precio, Fecha\_Realización, Fecha\_Terminación, DNI\_Empleado)

EMPLEADO (DNI, Dirección\_Calle, Dirección\_Número, Dirección\_Código\_Postal, Correo\_Electrónico, Teléfono)

La siguiente relación encontrada entre empleado y aplicación es la que tiene que ver con el desarrollo de estas últimas.

**Imagen 13. Entidad aplicación, entidad empleado, sus atributos y sus relaciones.**



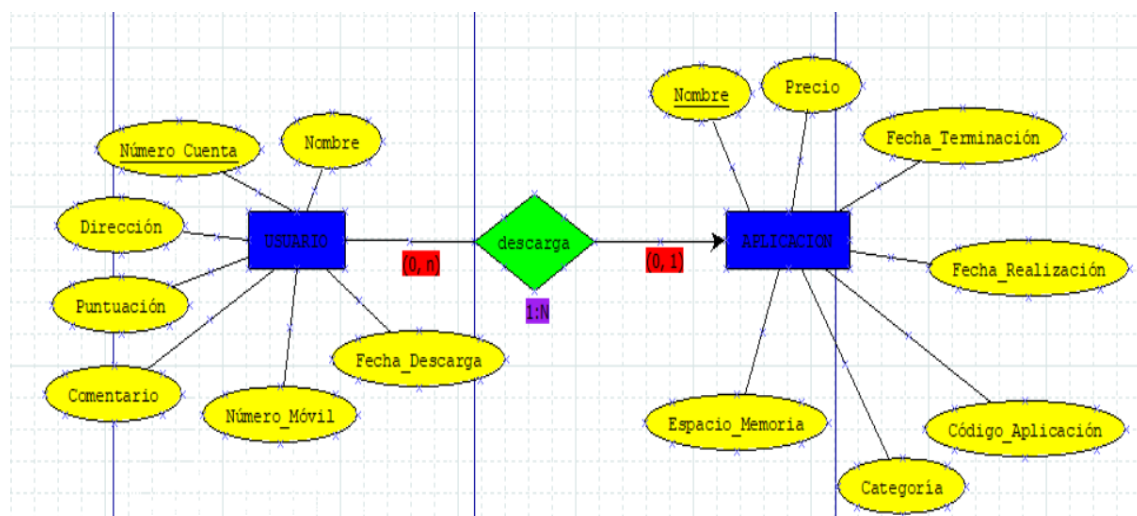
APLICACIÓN (Nombre, Código\_Aplicación, Categoría, Espacio\_Memoria, Precio, Fecha\_Realización, Fecha\_Terminación, DNI Empleado)

EMPLEADO (DNI, Dirección\_Calle, Dirección\_Número, Dirección\_Código\_Postal, Correo\_Electrónico, Teléfono)

DESARROLLA (DNI Empleado, Nombre Aplicación, Código Grupo)

La última relación encontrada, relaciona la entidad usuario y la entidad relación. Se trata de una relación de uno a muchos por lo que se colocará la clave primaria del lado uno en el lado de varios.

**Imagen 14. Entidad usuario, entidad aplicación, sus atributos y su relación.**



USUARIO (Número\_Cuenta, Nombre, Dirección, Puntuación, Comentario, Fecha\_Descarga, Número\_Móvil, Nombre\_Aplicación)

APLICACIÓN (Nombre, Código\_Aplicación, Categoría, Espacio\_Memoria, Precio, Fecha\_Realización, Fecha\_Terminación)

## IMPLEMENTACIÓN DEL PROYECTO EN MYSQL

Una vez definidas las tablas que formarán parte de la base de datos, se procede a la creación de ellas en el script del proyecto con MySQL Workbench.

El esquema general de implementación de la base de datos y sus correspondientes tablas se corresponde con lo que se explica a continuación, sin embargo, para conocer específicamente el nombre de la base de datos, el nombre de las tablas, los tipos de datos usados, las restricciones... es necesario consultar el script del proyecto donde se pueden encontrar estos detalles.

En primer lugar, se elimina la base de datos en caso de que ésta ya exista mediante el siguiente comando:

```
DROP database if exists TiendaAplicaciones;
```

Seguidamente, se crea la base de datos:

```
CREATE database TiendaAplicaciones;
```

```
USE TiendaAplicaciones;
```

Una vez creada la base de datos que contendrá las tablas, se crean las tablas dentro de la base de datos eliminándolas previamente en caso de que ya existan:

```
DROP table if exists tienda;
```

```
DROP table if exists empresa;
```

```
DROP table if exists empleado;
```

```
DROP table if exists aplicacion;
```

```
DROP table if exists usuario;
```

```
DROP table if exists trabaja;
```

```
DROP table if exists sube;
```

```
DROP table if exists desarrolla;
```

```

CREATE table tienda(
    nombre VARCHAR(50),
    paginaWeb VARCHAR(100),
    PRIMARY KEY(nombre)
);

CREATE table empresa(
    nombre VARCHAR(50),
    paisImpuestos VARCHAR(30) NOT NULL,
    correoElectronico VARCHAR(50),
    añoCreacion YEAR NOT NULL,
    paginaWeb VARCHAR(50),
    PRIMARY KEY(nombre)
);

CREATE table empleado(
    dni CHAR(9),
    direccionCalle VARCHAR(50) NOT NULL,
    direccionNumero VARCHAR(10) NOT NULL,
    direccionCP VARCHAR(10) NOT NULL,
    correoElectronico VARCHAR(50),
    telefono NUMERIC(9, 0),
    PRIMARY KEY(dni)
);

CREATE table aplicacion(
    nombreDeApp VARCHAR(50),
    codigoAplicacion CHAR(3) NOT NULL,
    categoria ENUM('Entretenimiento', 'Social', 'Educación') NOT NULL,
    espacioMemoria NUMERIC(5,2) NOT NULL,
    precio NUMERIC (4, 2) NOT NULL,
    fechaRealizacion DATE,
    fechaTerminacion DATE,

```

```

nombreEmpresa VARCHAR(50),
dniEmpleado CHAR(9),
PRIMARY KEY(nombreDeApp, nombreEmpresa, dniEmpleado),
FOREIGN KEY(nombreEmpresa) REFERENCES empresa(nombre)
ON DELETE cascade
ON UPDATE cascade,
FOREIGN KEY(dniEmpleado) REFERENCES empleado(dni)
ON DELETE restrict
ON UPDATE cascade
);
/*
En caso de borrar el nombre de una empresa, se borra la aplicación
En caso de actualizar el nombre de una empresa, se actualiza la aplicación
*****
***

En caso de borrar el DNI de un empleado, no se borra la aplicación
En caso de actualizar el DNI de un empleado, se actualiza la aplicación
*/

CREATE table usuario(
    numeroCuenta VARCHAR(30),
    nombreUser VARCHAR(50) NOT NULL,
    direccion VARCHAR(30) NOT NULL,
    puntuacion ENUM('1', '2', '3', '4', '5') NOT NULL,
    comentario VARCHAR(100),
    fechaDescarga DATE NOT NULL,
    numeroMovil NUMERIC(9,0),
    nombreAplicacion VARCHAR(50),
    PRIMARY KEY(numeroCuenta, nombreAplicacion),
    FOREIGN KEY(nombreAplicacion) REFERENCES
aplicacion(nombreDeApp)
ON DELETE cascade

```



```

        ON UPDATE cascade
    );
/*
En caso de borrar el nombre de una aplicación, se borra el usuario que la ha
descargado
En caso de actualizar el nombre de una aplicación, se actualiza el usuario que
la ha descargado
*/

```

```

CREATE table trabaja(
    nombreEmpresa VARCHAR(50),
    dniEmpleado CHAR(9),
    fechaContratacion DATE NOT NULL,
    fechaRenuncia DATE NOT NULL DEFAULT (CURRENT_DATE),
    PRIMARY KEY(nombreEmpresa, dniEmpleado, fechaContratacion),
    FOREIGN KEY(nombreEmpresa) REFERENCES empresa(nombre)
    ON DELETE cascade
    ON UPDATE cascade,
    FOREIGN KEY(dniEmpleado) REFERENCES empleado(dni)
    ON DELETE cascade
    ON UPDATE cascade
);
/*

```

En caso de borrar el nombre de una empresa, se borra el trabajador

En caso de actualizar el nombre de una empresa, se actualiza el trabajador

```

*****
***

```

En caso de borrar el DNI de un empleado, se borra el trabajador

En caso de actualizar el DNI de un empleado, se actualiza el trabajador

```

*/

```

```

CREATE table sube(
    nombreAplicacion VARCHAR(50),
    nombreTienda VARCHAR(50),

```

```

PRIMARY KEY(nombreAplicacion, nombreTienda),
FOREIGN KEY(nombreAplicacion) REFERENCES
aplicacion(nombreDeApp)
ON DELETE cascade
ON UPDATE cascade,
FOREIGN KEY(nombreTienda) REFERENCES tienda(nombre)
ON DELETE cascade
ON UPDATE cascade
);
/*

```

En caso de borrar el nombre de una aplicación, se borra la aplicación subida

En caso de actualizar el nombre de una aplicación, se actualiza la aplicación subida

```

*****
***

```

En caso de borrar el nombre de una tienda, se borra el nombre de la tienda

En caso de actualizar el nombre de una tienda, se actualiza el nombre de la tienda

```

*/

```

```

CREATE table desarrolla(
    dniEmpleado CHAR(9),
    nombreApp VARCHAR(50),
    codigoGrupo CHAR(1),
    PRIMARY KEY(dniEmpleado, nombreApp, codigoGrupo),
    FOREIGN KEY(dniEmpleado) REFERENCES empleado(dni)
    ON DELETE restrict
    ON UPDATE cascade,
    FOREIGN KEY(nombreApp) REFERENCES aplicacion(nombreDeApp)
    ON DELETE cascade
    ON UPDATE cascade
);
/*

```

En caso de borrar el DNI de un empleado, no se borra el grupo que desarrolla la aplicación

En caso de actualizar el DNI de un empleado, se actualiza el grupo que desarrolla la aplicación

\*\*\*\*\*  
\*\*\*

En caso de borrar el nombre de una aplicación, se borra el grupo que desarrolla la aplicación

En caso de actualizar el nombre de una aplicación, se actualiza el grupo que desarrolla la aplicación

\*/

A continuación, se crean las instancias que forman parte de cada tupla de la tabla mediante el siguiente comando:

-- Se comienzan a crear las instancias que forman las tuplas de la tabla tienda

```
INSERT INTO tienda VALUES('App Store', 'https://www.apple.com/es/app-store/');
```

```
INSERT INTO tienda VALUES('Google Play Store', 'https://play.google.com/work?hl=es&gl=US');
```

```
INSERT INTO tienda VALUES('App World', NULL);
```

```
INSERT INTO tienda VALUES('Market Place', NULL);
```

```
INSERT INTO tienda VALUES('OVITienda', NULL);
```

```
INSERT INTO tienda VALUES('AppCatalog', NULL);
```

```
INSERT INTO tienda VALUES('Appstore', 'https://www.amazon.es/mobile-apps/b?ie=UTF8&node=1661649031');
```

-- Se comienzan a crear las instancias que forman las tuplas de la tabla empresa

```
INSERT INTO empresa VALUES('Gameberry Labs Private Limited', 'India', NULL, 2017, 'http://gameberrylabs.com/');
```

```
INSERT INTO empresa VALUES('Rovio Entertainment Corporation', 'Finlandia', 'support@rovio.com', 2017, 'https://www.rovio.com/');
```

```
INSERT INTO empresa VALUES('BP Mobile LLC', 'China', 'support@bpmobile.com', 2016, 'https://bpmobile.com/');
```

```
INSERT INTO empresa VALUES('Supercell', 'Finlandia', 'legal-requests@supercell.com', 2010, 'https://supercell.com/');
```

```
INSERT INTO empresa VALUES('Adevinta Spain S.L.U.', 'España', NULL, 2002, 'https://adevinta.es/');
```

-- Se comienzan a crear las instancias que forman la tuplas de la tabla empleado

```
INSERT INTO empleado VALUES('41298756Q', 'Calle Bonaire', '12', '32096', 'prgomez@gmail.com', 903689512);
```

```
INSERT INTO empleado VALUES('87302912R', 'Calle Diderot', '6', '87521', 'immartos@gmail.com', 904873200);
```

```
INSERT INTO empleado VALUES('42659852T', 'Calle Gálica', '18', '27543', 'namartin@gmail.com', 907896201);
```

```
INSERT INTO empleado VALUES('96312578Y', 'Calle Doctor Buades', '75', '45263', 'ajgaspar@gmail.com', 909521304);
```

```
INSERT INTO empleado VALUES('43089650P', 'Calle Rozas', '51', '25816', 'bugavilan@gmail.com', 912589209);
```

```
INSERT INTO empleado VALUES('85321976L', 'Calle Aviación', '13', '85234', 'cvmota@gmail.com', 903813655);
```

```
INSERT INTO empleado VALUES('72001139A', 'Avenida Tribunal', '2', '63002', 'dzvara@gmail.com', 906032002);
```

```
INSERT INTO empleado VALUES('99632504G', 'Calle Oporto', '33', '32018', 'emgarcia@gmail.com', 904556217);
```

```
INSERT INTO empleado VALUES('45200326V', 'Calle Torcal', '25', '74300', 'frrueda@gmail.com', 912339856);
```

```
INSERT INTO empleado VALUES('30302153H', 'Avenida América', '10', '12321', 'ghgutierrez@gmail.com', 906523474);
```

```
INSERT INTO empleado VALUES('96325478B', 'Calle Moncloa', '36', '85103', 'hdpazos@gmail.com', 907584630);
```

```
INSERT INTO empleado VALUES('45219875D', 'Calle Victoria', '87', '74203', 'jgfuentes@gmail.com', 901101217);
```

```
INSERT INTO empleado VALUES('63200040Z', 'Calle Norte', '3', '32001', 'lmuñoz@gmail.com', 906235491);
```

```
INSERT INTO empleado VALUES('33278994S', 'Avenida San Juan', '15', '28763', 'rlbarrio@gmail.com', 987632541);
```

```
INSERT INTO empleado VALUES('11330862L', 'Calle Rosal', '11', '12640', 'cpaguilar@gmail.com', 962103647);
```

-- Se comienzan a crear las instancias que forman la tuplas de la tabla aplicación

```
INSERT INTO aplicacion VALUES('InfoJobs', '001', 'Social', 40.1, 29.99, '1998-11-02', '2000-02-17', 'Adevinta Spain S.L.U.', '41298756Q');
```

```
INSERT INTO aplicacion VALUES('Hay Day', '002', 'Entretenimiento', 204.6, 9.99, '2014-05-05', '2015-01-19', 'Supercell', '87302912R');
```

```
INSERT INTO aplicacion VALUES('iScanner', '003', 'Educación', 187.1, 15.90, '2017-02-15', '2017-12-06', 'BP Mobile LLC', '42659852T');
```

```
INSERT INTO aplicacion VALUES('Angry Birds 2', '004', 'Entretenimiento', 275.6, 39.99, '2019-07-11', '2020-02-03', 'Rovio Entertainment Corporation', '96312578Y');
```

```
INSERT INTO aplicacion VALUES('Parchis STAR', '005', 'Entretenimiento', 84, 6.99, '2020-02-05', '2020-03-17', 'Gameberry Labs Private Limited', '43089650P');
```

```
INSERT INTO aplicacion VALUES('Milanuncios', '006', 'Social', 62.9, 19.99, '2005-01-11', '2005-09-07', 'Adevinta Spain S.L.U.', '11330862L');
```

```
INSERT INTO aplicacion VALUES('Clash of Clans', '007', 'Entretenimiento', 321.4, 50.00, '2012-06-05', '2012-09-02', 'Supercell', '33278994S');
```

```
INSERT INTO aplicacion VALUES('Fax App', '008', 'Educación', 43.5, 17.59, '2018-12-14', '2019-02-09', 'BP Mobile LLC', '42659852T');
```

```
INSERT INTO aplicacion VALUES('Darkfire Heroes', '009', 'Entretenimiento', 525.4, 4.90, '2020-09-11', '2021-04-02', 'Rovio Entertainment Corporation', '30302153H');
```

```
INSERT INTO aplicacion VALUES('Ludo STAR', '010', 'Entretenimiento', 83.9, 2.39, '2018-11-08', '2019-02-11', 'Gameberry Labs Private Limited', '63200040Z');
```

```
INSERT INTO aplicacion VALUES('Safe24', '011', 'Social', 55.3, 12.59, '2016-10-18', '2017-03-06', 'BP Mobile LLC', '72001139A');
```

-- Se comienzan a crear las instancias que forman las tuplas de la tabla usuario

```
INSERT INTO usuario VALUES('ES0213586321400297855620', 'Jimena Casillas Salgado', 'España', 4, 'Muy divertida', '2020-03-31', NULL, 'Hay Day');
```

```
INSERT INTO usuario VALUES('ES0213586321400297855620', 'Jimena Casillas Salgado', 'España', 2, NULL, '2020-11-04', NULL, 'InfoJobs');
```

```
INSERT INTO usuario VALUES('ES7854100369885554102037', 'Alberto Camacho Alonso', 'España', 4, 'Una aplicación útil y rápida', '2019-05-06', NULL, 'iScanner');
```

INSERT INTO usuario VALUES('PT5298665330002157446953', 'João Teixeira Gonçalves', 'Portugal', 3, NULL, '2021-06-27', NULL, 'Angry Birds 2');

INSERT INTO usuario VALUES('FR0300988875201395420117', 'Antoine Lenglet Lloris', 'Francia', 5, NULL, '2020-03-31', NULL, 'Parchis STAR');

INSERT INTO usuario VALUES('PT8503226548954301247865', 'Hugo Ferreira Reis', 'Portugal', 4, NULL, '2019-05-06', NULL, 'Clash of Clans');

INSERT INTO usuario VALUES('ES0213586321400297855620', 'Jimena Casillas Salgado', 'España', 5, 'Una app inmejorable', '2019-05-06', NULL, 'Milanuncios');

INSERT INTO usuario VALUES('PT5298665330002157446953', 'João Teixeira Gonçalves', 'Portugal', 1, NULL, '2019-05-06', NULL, 'Fax App');

INSERT INTO usuario VALUES('PT0213587454462139798445', 'Miguel Barbosa Moutinho', 'Portugal', 5, NULL, '2020-03-31', NULL, 'Darkfire Heroes');

INSERT INTO usuario VALUES('FR0300988875201395420117', 'Adrien Toulalan Durand', 'Francia', 1, NULL, '2019-05-06', NULL, 'Ludo STAR');

INSERT INTO usuario VALUES('FR0300988875201395420117', 'Antoine Lenglet Lloris', 'Francia', 2, NULL, '2021-06-27', NULL, 'Clash of Clans');

INSERT INTO usuario VALUES('PT0213587454462139798445', 'Miguel Barbosa Moutinho', 'Portugal', 4, NULL, '2021-06-27', NULL, 'Fax App');

INSERT INTO usuario VALUES('FR0300988875201395420117', 'Adrien Toulalan Durand', 'Francia', 3, NULL, '2020-03-31', NULL, 'Angry Birds 2');

INSERT INTO usuario VALUES('IT1002301245789544233321', 'Laura Ferrari Conte', 'Italia', 5, NULL, '2019-05-06', NULL, 'Hay Day');

INSERT INTO usuario VALUES('PT5203012455877796542102', 'Telma Almeida Pinto', 'Portugal', 3, NULL, '2020-11-04', NULL, 'Clash of Clans');

INSERT INTO usuario VALUES('ES1058667530019756644108', 'Carmen Delgado Estrada', 'España', 3, 'He encontrado trabajo a través de esta app', '2020-03-31', NULL, 'Milanuncios');

INSERT INTO usuario VALUES('ES0021344875266479632541', 'José Herrera Pombo', 'España', 4, 'Genial para pasar mi tiempo libre', '2020-03-31', NULL, 'Clash of Clans');

INSERT INTO usuario VALUES('PT6201245877754621315100', 'Estela Oliveira Lobo', 'Portugal', 4, NULL, '2020-03-31', NULL, 'Hay Day');

INSERT INTO usuario VALUES('FR5821420032158785542625', 'Camille Dubois Laurent', 'Francia', 5, NULL, '2020-11-04', NULL, 'Clash of Clans');

INSERT INTO usuario VALUES('PT8503226548954301247865', 'Hugo Ferreira Reis', 'Portugal', 4, NULL, '2019-05-06', NULL, 'Safe24');

```

INSERT INTO usuario VALUES('ES0029874552130021201548', 'Pablo
Bermejo Díaz', 'España', 4, NULL, '2020-03-31', 652895321, 'InfoJobs');

INSERT INTO usuario VALUES('IT5210231254447854563206', 'Andrea Rossi
Esposito', 'Italia', 4, NULL, '2021-06-27', 253695247, 'Hay Day');

INSERT INTO usuario VALUES('IT5210325887899655455128', 'Rosa Conti
Bianco', 'Italia', 3, NULL, '2021-08-03', 242658790, 'Clash of Clans');

INSERT INTO usuario VALUES('PT8503226548954301247865', 'Hugo Ferreira
Reis', 'Portugal', 4, NULL, '2020-11-04', 962548703, 'Hay Day');

INSERT INTO usuario VALUES('IT5210325887899655455128', 'Rosa Conti
Bianco', 'Italia', 5, NULL, '2020-11-04', 242658790, 'Angry Birds 2');

```

-- Se comienzan a crear las instancias que forman las tuplas de la tabla trabaja

```

INSERT INTO trabaja VALUES('Gameberry Labs Private Limited', '41298756Q',
'2018-02-01', DEFAULT);

INSERT INTO trabaja VALUES('Gameberry Labs Private Limited', '87302912R',
'2018-09-01', DEFAULT);

INSERT INTO trabaja VALUES('Gameberry Labs Private Limited', '42659852T',
'2020-10-01', DEFAULT);

INSERT INTO trabaja VALUES('Gameberry Labs Private Limited', '99632504G',
'2019-09-01', DEFAULT);

INSERT INTO trabaja VALUES('Rovio Entertainment Corporation',
'96312578Y', '2020-06-01', DEFAULT);

INSERT INTO trabaja VALUES('Rovio Entertainment Corporation',
'43089650P', '2017-07-01', DEFAULT);

INSERT INTO trabaja VALUES('Rovio Entertainment Corporation', '85321976L',
'2020-05-01', DEFAULT);

INSERT INTO trabaja VALUES('Rovio Entertainment Corporation',
'30302153H', '2019-06-01', DEFAULT);

INSERT INTO trabaja VALUES('BP Mobile LLC', '72001139A', '2016-09-01',
DEFAULT);

INSERT INTO trabaja VALUES('BP Mobile LLC', '30302153H', '2017-01-01',
'2019-01-30');

INSERT INTO trabaja VALUES('BP Mobile LLC', '99632504G', '2018-03-01',
'2019-04-30');

INSERT INTO trabaja VALUES('BP Mobile LLC', '45200326V', '2019-08-01',
DEFAULT);

```

```

INSERT INTO trabaja VALUES('BP Mobile LLC', '96325478B', '2016-02-01',
DEFAULT);

INSERT INTO trabaja VALUES('BP Mobile LLC', '42659852T', '2017-10-01',
'2018-11-30');

INSERT INTO trabaja VALUES('BP Mobile LLC', '96312578Y', '2012-02-01',
'2017-05-30');

INSERT INTO trabaja VALUES('Supercell', '30302153H', '2011-03-01', '2016-
08-30');

INSERT INTO trabaja VALUES('Supercell', '96325478B', '2014-07-01', '2015-
09-30');

INSERT INTO trabaja VALUES('Supercell', '45219875D', '2015-04-01',
DEFAULT);

INSERT INTO trabaja VALUES('Supercell', '63200040Z', '2012-07-01',
DEFAULT);

INSERT INTO trabaja VALUES('Supercell', '96312578Y', '2018-02-01', '2019-
03-30');

INSERT INTO trabaja VALUES('Supercell', '41298756Q', '2011-01-01', '2018-
01-30');

INSERT INTO trabaja VALUES('Supercell', '43089650P', '2012-07-01', '2014-
11-30');

INSERT INTO trabaja VALUES('Adevinta Spain S.L.U.', '63200040Z', '2003-03-
01', '2011-05-30');

INSERT INTO trabaja VALUES('Adevinta Spain S.L.U.', '33278994S', '1998-02-
01', '2015-09-01');

INSERT INTO trabaja VALUES('Adevinta Spain S.L.U.', '11330862L', '1999-04-
01', DEFAULT);

INSERT INTO trabaja VALUES('Adevinta Spain S.L.U.', '33278994S', '2017-09-
01', DEFAULT);

INSERT INTO trabaja VALUES('Adevinta Spain S.L.U.', '72001139A', '2003-04-
01', '2016-07-30');

INSERT INTO trabaja VALUES('Adevinta Spain S.L.U.', '43089650P', '2004-07-
01', '2010-09-30');

INSERT INTO trabaja VALUES('Adevinta Spain S.L.U.', '30302153H', '2005-04-
01', '2006-12-30');

INSERT INTO trabaja VALUES('Adevinta Spain S.L.U.', '85321976L', '2004-02-
01', '2006-05-30');

```



-- Se comienzan a crear las instancias que forman las tuplas de la tabla sube

```
INSERT INTO sube VALUES('InfoJobs', 'App Store');
INSERT INTO sube VALUES('InfoJobs', 'Google Play Store');
INSERT INTO sube VALUES('InfoJobs', 'AppCatalog');
INSERT INTO sube VALUES('InfoJobs', 'Appstore');
INSERT INTO sube VALUES('Hay Day', 'App Store');
INSERT INTO sube VALUES('Hay Day', 'Google Play Store');
INSERT INTO sube VALUES('Hay Day', 'Market Place');
INSERT INTO sube VALUES('Hay Day', 'OVITienda');
INSERT INTO sube VALUES('iScanner', 'App Store');
INSERT INTO sube VALUES('iScanner', 'Google Play Store');
INSERT INTO sube VALUES('iScanner', 'Market Place');
INSERT INTO sube VALUES('Angry Birds 2', 'App Store');
INSERT INTO sube VALUES('Angry Birds 2', 'Google Play Store');
INSERT INTO sube VALUES('Angry Birds 2', 'Market Place');
INSERT INTO sube VALUES('Angry Birds 2', 'App World');
INSERT INTO sube VALUES('Parchis STAR', 'App Store');
INSERT INTO sube VALUES('Parchis STAR', 'Google Play Store');
INSERT INTO sube VALUES('Parchis STAR', 'Market Place');
INSERT INTO sube VALUES('Parchis STAR', 'OVITienda');
INSERT INTO sube VALUES('Parchis STAR', 'AppCatalog');
INSERT INTO sube VALUES('Milanuncios', 'App Store');
INSERT INTO sube VALUES('Milanuncios', 'Google Play Store');
INSERT INTO sube VALUES('Clash of Clans', 'App Store');
INSERT INTO sube VALUES('Clash of Clans', 'Google Play Store');
INSERT INTO sube VALUES('Clash of Clans', 'Market Place');
INSERT INTO sube VALUES('Clash of Clans', 'OVITienda');
INSERT INTO sube VALUES('Clash of Clans', 'AppCatalog');
INSERT INTO sube VALUES('Clash of Clans', 'Appstore');
INSERT INTO sube VALUES('Clash of Clans', 'App World');
```

```

INSERT INTO sube VALUES('Fax App', 'App Store');
INSERT INTO sube VALUES('Darkfire Heroes', 'App Store');
INSERT INTO sube VALUES('Darkfire Heroes', 'Google Play Store');
INSERT INTO sube VALUES('Darkfire Heroes', 'AppCatalog');
INSERT INTO sube VALUES('Darkfire Heroes', 'Appstore');
INSERT INTO sube VALUES('Darkfire Heroes', 'App World');
INSERT INTO sube VALUES('Ludo STAR', 'App Store');
INSERT INTO sube VALUES('Ludo STAR', 'Google Play Store');
INSERT INTO sube VALUES('Ludo STAR', 'Market Place');
INSERT INTO sube VALUES('Safe24', 'App Store');
INSERT INTO sube VALUES('Safe24', 'Google Play Store');

```

-- Se comienzan a crear las instancias que forman las tuplas de la tabla sube

```

INSERT INTO desarrolla VALUES('11330862L', 'InfoJobs', 'A');
INSERT INTO desarrolla VALUES('33278994S', 'InfoJobs', 'A');
INSERT INTO desarrolla VALUES('63200040Z', 'Hay Day', 'B');
INSERT INTO desarrolla VALUES('45219875D', 'Hay Day', 'B');
INSERT INTO desarrolla VALUES('96325478B', 'iScanner', 'C');
INSERT INTO desarrolla VALUES('30302153H', 'Angry Birds 2', 'D');
INSERT INTO desarrolla VALUES('45200326V', 'Angry Birds 2', 'D');
INSERT INTO desarrolla VALUES('99632504G', 'Parchis STAR', 'E');
INSERT INTO desarrolla VALUES('72001139A', 'Milanuncios', 'F');
INSERT INTO desarrolla VALUES('85321976L', 'Milanuncios', 'F');
INSERT INTO desarrolla VALUES('43089650P', 'Clash of Clans', 'G');
INSERT INTO desarrolla VALUES('42659852T', 'Fax App', 'H');
INSERT INTO desarrolla VALUES('30302153H', 'Darkfire Heroes', 'I');
INSERT INTO desarrolla VALUES('87302912R', 'Ludo STAR', 'J');
INSERT INTO desarrolla VALUES('96312578Y', 'Safe24', 'K');

```

## **CONSULTAS SOBRE LA BASE DE DATOS**

1. Se precisa conocer la fecha en la que se han realizado más descargas.

```
SELECT distinct(fechaDescarga), count(fechaDescarga) as cantidadDescargas  
FROM usuario  
GROUP BY fechaDescarga  
ORDER BY cantidadDescargas DESC;
```

2. Se precisa conocer a qué país pertenecen los usuarios que realizan más descargas.

```
SELECT distinct(direccion), count(direccion) as cantidadDescargas  
FROM usuario  
GROUP BY direccion  
ORDER BY cantidadDescargas DESC;
```

3. Se precisa conocer cuál es la puntuación media de cada una de las apps.

```
SELECT distinct(nombreAplicacion), avg(puntuacion) as puntuacionMedia  
FROM usuario  
GROUP BY nombreAplicacion  
ORDER BY puntuacionMedia ASC;
```

4. Se desea conocer qué usuarios han descargado dos o más de las aplicaciones existentes.

```
SELECT distinct(nombreUser) as nombreUsuario, count(nombreAplicacion) as  
cantidadDescargas  
FROM usuario  
GROUP BY nombreUsuario  
HAVING cantidadDescargas >= 2  
ORDER BY cantidadDescargas DESC;
```

5. Se necesita conocer el total de espacio memoria en megabytes, de menor a mayor, de todas las aplicaciones descargadas por cada usuario.

```
SELECT nombreUser as nombre, sum(espacioMemoria) as
espacioMemoriaTotal

FROM usuario INNER JOIN aplicacion ON aplicacion.nombreDeApp =
usuario.nombreAplicacion

GROUP BY nombreUser

ORDER BY espacioMemoriaTotal ASC;
```

6. Se desea saber aquellos empleados de empresas que tengan una avenida y un código postal impar como dirección.

```
SELECT dni, direccionCalle

FROM empleado

WHERE direccionCalle LIKE '%Avenida%' AND (direccionCP % 2) <> 0;
```

7. Se precisa conocer los datos de aquellos empleados que no dirigen ninguna aplicación para contactarles a través de correo electrónico o mediante número de teléfono.

```
SELECT dni, correoElectronico, telefono

FROM empleado LEFT JOIN aplicacion ON empleado.dni =
aplicacion.dniEmpleado

WHERE aplicacion.dniEmpleado IS NULL;
```

8. Se necesita saber el nombre de los usuarios que han descargado aplicaciones de la categoría 'social' y la puntuación media que le han dado a esas aplicaciones.

```
SELECT nombreUser as nombreUsuario, avg(puntuacion) as puntuacionMedia

FROM usuario RIGHT JOIN aplicacion ON usuario.nombreAplicacion =
aplicacion.nombreDeApp

WHERE aplicacion.categoria = 'Social'

GROUP BY nombreUser;
```

9. Se desea obtener una lista con los empleados que cuentan con una mayor experiencia laboral en el sector hasta los que tienen menos años de experiencia

```
SELECT dniEmpleado, sum(TIMESTAMPDIFF(YEAR, fechaContratacion,  
fechaRenuncia) as experienciaProfesional
```

```
FROM trabaja
```

```
GROUP BY dniEmpleado
```

```
ORDER BY experienciaProfesional DESC;
```

10. Se precisa saber el nombre y la página web de la/s empresa/s que han realizado más de dos aplicaciones.

```
SELECT nombre, paginaWeb
```

```
FROM empresa INNER JOIN aplicacion ON empresa.nombre =  
aplicacion.nombreEmpresa
```

```
GROUP BY nombre
```

```
HAVING count(aplicacion.nombreDeApp) > 2;
```

11. Es necesario conocer qué usuarios de aplicaciones han gastado 50 euros en alguna aplicación y que, además, tengan España como país de dirección.

```
CREATE view gastoUser (nombreUsuario, pais) as
```

```
    SELECT nombreUser, direccion
```

```
    FROM usuario LEFT JOIN aplicacion ON usuario.nombreAplicacion =  
    aplicacion.nombreDeApp
```

```
    WHERE aplicacion.precio = 50 AND usuario.direccion = 'España';
```

```
SELECT *
```

```
FROM gastoUser;
```

12. Se desea conocer qué aplicaciones están subidas a la tienda OVITienda, a la tienda Market Place o simultáneamente a las dos tiendas.

```
(SELECT nombreAplicacion
FROM sube RIGHT JOIN tienda ON sube.nombreTienda = tienda.nombre
WHERE nombreTienda = 'OVITienda')
UNION
(SELECT nombreAplicacion
FROM sube RIGHT JOIN tienda ON sube.nombreTienda = tienda.nombre
WHERE nombreTienda = 'Market Place');
```

13. Se necesita conocer qué grupos han desarrollado aplicaciones que contengan la palabra 'STAR' en el nombre de la aplicación.

```
CREATE view coste_total (id_grupo, app) as
    SELECT codigoGrupo, nombreAPP
    FROM desarrolla LEFT JOIN aplicacion ON desarrolla.nombreApp =
aplicacion.nombreDeApp
    GROUP BY nombreApp
    HAVING desarrolla.nombreApp LIKE '%STAR%';

SELECT id_grupo
FROM coste_total;
```

14. Es necesario conocer qué empleados han trabajado en alguna empresa que paga impuestos en Finlandia y cuyo año de creación es anterior a 2014

```
SELECT dniEmpleado
FROM trabaja RIGHT JOIN empresa ON trabaja.nombreEmpresa =
empresa.nombre
WHERE empresa.paisImpuestos = 'Finlandia' AND empresa.añoCreacion <
2014;
```

15. Se precisa conocer los empleados que trabajan actualmente y en qué empresa lo hacen

```
SELECT dniEmpleado, nombreEmpresa
FROM trabaja LEFT JOIN empresa ON trabaja.nombreEmpresa =
empresa.nombre
WHERE fechaRenuncia = CURRENT_DATE();
```

16. La empresa que ha desarrollado la aplicación Clash of Clans ha decidido bajar el precio de la aplicación en nueve euros

```
USE TiendaAplicaciones;
DROP table if exists updApp;
CREATE table updApp(
    nomApp VARCHAR(50),
    precioAntiguo NUMERIC (4, 2),
    precioNuevo NUMERIC (4, 2),
    fechaActualizacion DATETIME,
    usuario VARCHAR(30)
);

delimiter //
CREATE TRIGGER actualizaApp BEFORE UPDATE ON aplicacion
FOR EACH ROW
BEGIN
    INSERT INTO updApp VALUES(old.nombreDeApp, old.precio,
new.precio, now(), current_user());
END//

UPDATE aplicacion
SET precio = precio - 9.00
WHERE nombreDeApp = 'Clash of Clans';
```