

1.- Descripción de la tarea

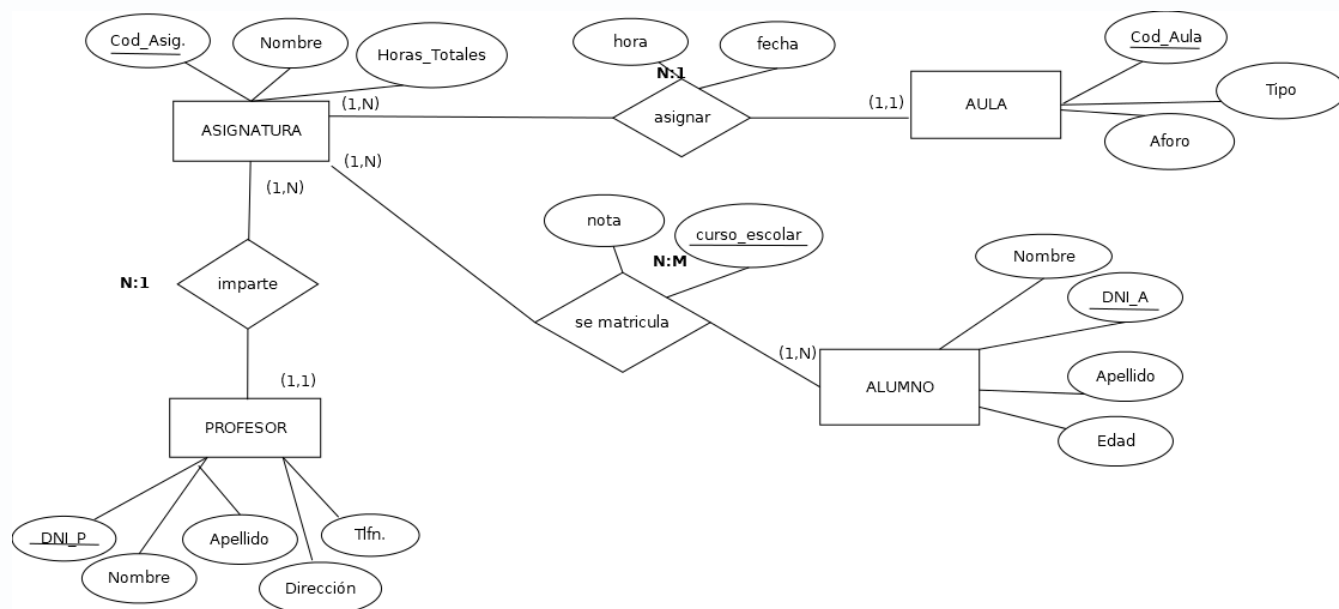
Caso práctico

Adrián lleva muchos años trabajando con bases de datos. **Lucía** trabaja en su equipo y le ha pasado el diagrama Entidad Relación y el modelo relacional con el que tiene que trabajar para implementar las tablas en MySQL. Tiene que utilizar el lenguaje SQL para generar las sentencias de creación de tablas cumpliendo con los requisitos que le han especificado. Debe elegir bien los tipos de datos así como los tamaños correctos de los campos para que la BD sea lo más eficiente posible.

¿Qué te pedimos que hagas?

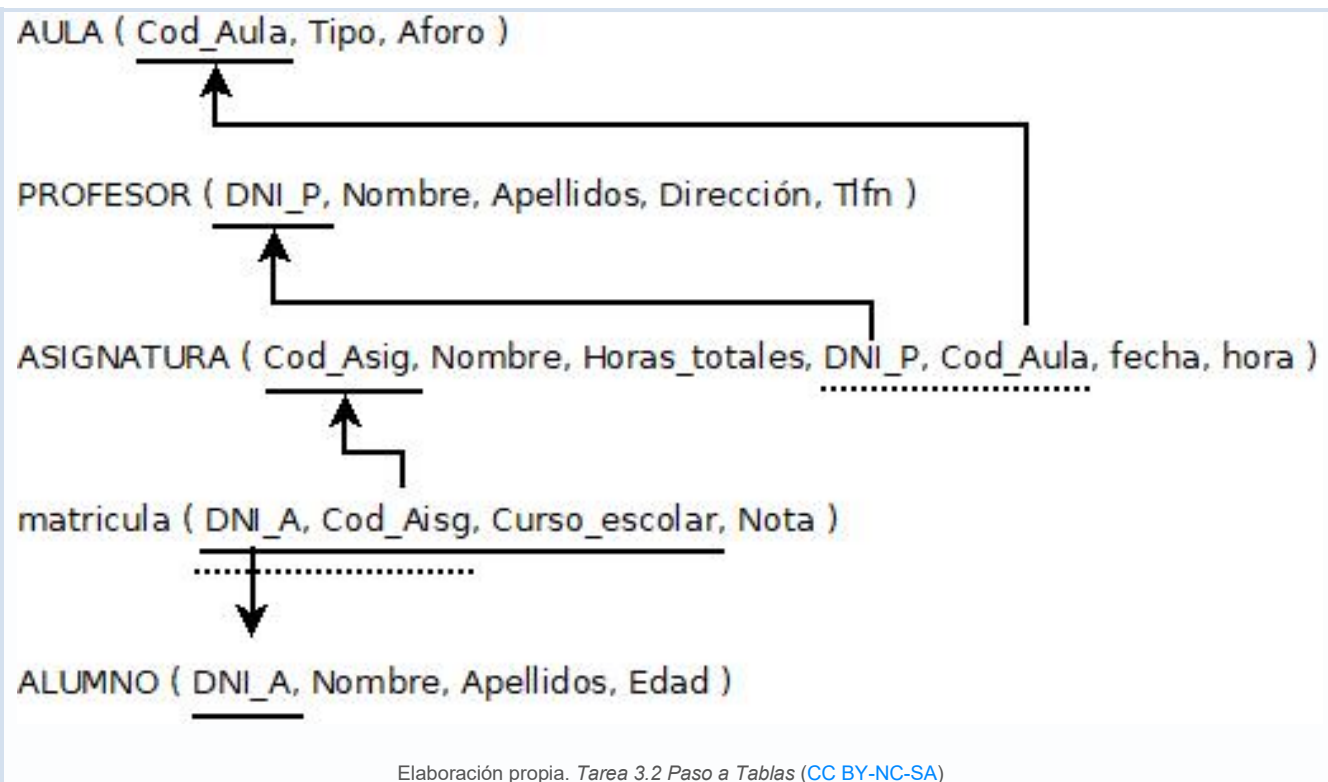
La dirección de un instituto nos ha pedido que realicemos una base de datos para controlar los profesores, asignaturas, alumnos y aulas que van a tener este año. Después de haber realizado el análisis de todos los requerimientos que nos han pedido, se ha llegado al siguiente diagrama E-R y al Modelo Relacional correspondiente. Ahora, es el momento de implementar con MySQL dichas tablas y por tanto realizar un script con lenguaje SQL para que funcione perfectamente en dicho SGBD.

DIAGRAMA E-R



Elaboración propia. Tarea 3.1 Diagrama (CC BY-NC-SA)

MODELO RELACIONAL



Partiendo de dichas tablas, debes realizar los siguientes apartados:

A) Escribe las sentencias de creación de tablas en el orden apropiado utilizando el lenguaje SQL para que funcionen correctamente en MySQL. Debes tener en cuenta lo siguiente:

- Elegir el **tipo de dato MySQL** y el **tamaño** más adecuado para cada campo teniendo en cuenta los valores que éstos pueden almacenar.
- Recuerda nombrar adecuadamente los **atributos, tablas y restricciones** teniendo en cuenta las reglas que se explican en los contenidos de la unidad.
- Se debe **cumplir la regla de integridad referencial** teniendo en cuenta que **para cualquier clave ajena cuando un valor de la tabla principal se elimine, se pondrán a NULL los valores a los que referencie**.
- Incluir en las sentencias de creación de tablas las siguientes restricciones utilizando el lenguaje SQL:
 1. Los campos **Cod_Asig**, **Cod_Aula**, **DNI_A** y **DNI_P** identifican de forma única cada registro de sus tablas correspondientes y están formados por una cadena de caracteres alfanuméricos de longitud fija
 2. Los campos **DNI_A** y **Cod_Asig** de la tabla Se_matricula provienen de las tablas de ALUMNO y ASIGNATURA mediante las relaciones correspondientes.
 3. El valor del campo **Aforo** en la tabla AULA por defecto debe ser 30.
 4. El campo **nombre** de la tabla ASIGNATURA no puede tener valores duplicados.

5. El campo **horas_totales** en la tabla ASIGNATURA no puede tomar valores nulos.

B) Después de crear las tablas, vamos a realizar algunas modificaciones sobre ellas. Escribe las sentencias SQL apropiadas para cada uno de los subapartados siguientes:

1. Eliminar el campo **edad** de la tabla ALUMNO.
2. Crear un índice en el campo **nota** de la tabla Se_matricula.
3. Añadir el campo **estudios** en la tabla PROFESOR y que éste sólo pueda tomar los valores "Diplomado", "Licenciado" o "Catedrático".
4. Crear el usuario "**director**" con la contraseña "**admin**" para que se pueda conectar desde cualquier equipo.
5. Conceder los privilegios de actualizar y borrar datos al usuario "**secretaria**" en la tabla ASIGNATURA.
6. Quitar el privilegio para eliminar datos en la tabla ASIGNATURA al usuario "**secretaria**".

A) Escribe las sentencias de creación de tablas **en el orden apropiado** utilizando el lenguaje SQL **para que funcionen correctamente en MySQL**.

1. Pues en primer lugar voy a crear la base de datos:

```
CREATE DATABASE Instituto;
```

2. Una vez creada la base de datos comenzare con la creación de las tablas:

- Tabla Asignaturas, correspondiente a la entidad asignatura con los atributos [(Cod_Asig, Nombre, Horas_Totales), además de las claves foráneas (DNI_P, Cod_Aula, fecha, hora)];

```
CREATE TABLE INSTITUTO. Asignaturas (
```

```
Cod_Asig CHAR (8) NOT NULL UNIQUE PRIMARY KEY,
```

```
Nombre VARCHAR (45) UNIQUE,
```

```
Horas_Totales TIME (839)* NOT NULL),
```

```
Fecha DATE,
```

```
Hora TIME (24) *me sale para ponerlo, pero luego se quita
```

```
DNI_P CHAR (9)
```

```
Cod_Aula CHAR (8)
```

```
CONSTRAINT Profesores, Aulas_Cod_F_FK FOREIGN KEY (DNI_P, Cod_Aula)
```

```
REFERENCES Profesores, Aulas(DNI_P ,Cod_Aula) ON DELETE SET NULL); /*este  
proceso en MySQL lo haremos de la siguiente forma:*/
```

*Pues **crearemos las columnas DNI_P, Cod_Aula**:

Luego entre las pestañas que tenemos en MySQL para intercambiar entre columnas, Indexes, Foreign Key, etc... **Vamos a seleccionar la pestaña Foreign Key** y **vamos a introducir las claves y las tablas de procedencia**, y lo vamos a relacionar todo.

1.-Bien primero campo Foreign Key: DNI_P

2.-Segundo vamos a completar el campo Reference Table: Profesores

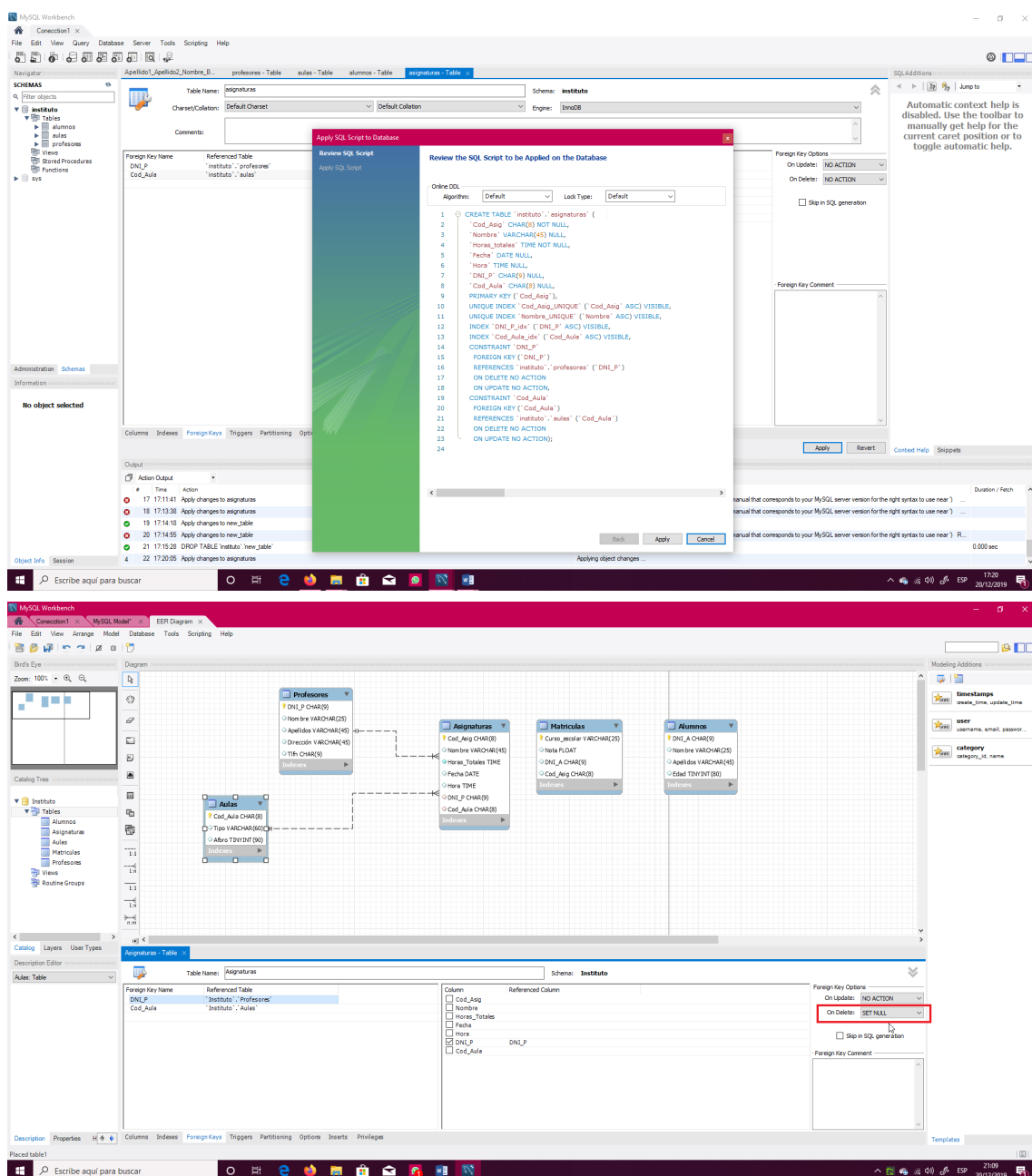
3.-Tercero voy a realizar la relación de esta tabla 'Asignaturas' que va a llamar a la tabla 'Profesores' mediante la clave Foránea 'DNI_P' (y esto es importante para que la relación sea la correcta según el orden en el que se realiza); además ha de tener

esta clave Foránea el mismo tipo de dato en ambas tablas es decir seria CHAR (9) en ambas tablas.

4.-Cuarto, siguiendo las reglas de integridad referencial, pues con la opción ON DELETE SET NULL, haremos que se establezca a NULL el valor de la clave secundaria cuando se elimina el registro en la tabla principal o se modifica el valor del campo referenciado;

REFERENCES Profesores, Aulas(DNI_P ,Cod_Aula) ON DELETE SET NULL;

-Será el mismo procedimiento para cada clave Foránea, por lo tanto, una vez explicado está, las demás las voy a realizar sin más que añadir.



- Tabla Profesores, correspondiente a la entidad profesor con los atributos (DNI_P, Nombre, Apellidos, Dirección, Tlfn);

CREATE TABLE INSTITUTO. Profesores (

DNI_P CHAR (9) UNIQUE PRIMARY KEY,

Nombre VARCHAR (25),

Apellido VARCHAR (45),

Dirección VARCHAR (45),

Tlfn CHAR (9)); /*en principio puse TINYINT porque era numérico, pero recapacite y creo será mejor un CHAR (9) de 9 bit ya que tanto tlfn móvil o fijo, contienen 9 dígitos o números.*/*

- Tabla Aulas, correspondiente a la entidad aula con los atributos (Cod_Aula, Tipo, Aforo);

CREATE TABLE INSTITUTO. Aulas (

Cod_Aula CHAR (8) NOT NULL UNIQUE PRIMARY KEY,

Tipo VARCHAR (60)

Aforo TINYINT (2) DEFAULT (30),

ENGINE=InnoDB; /*está columna no la tengo que insertar ya que al hacerlo por el MySQL Workbench a la hora de crear las tablas ya viene seleccionado en una pestaña. (Lo menciono porque creo si tuviera que hacerlo de otro modo (modo comandos) si tendría que escribir este comando o columna en todas las tablas).*/

- Tabla Alumnos, correspondiente a la entidad alumno con los atributos (DNI_A, Nombre, Apellidos, Edad);

CREATE TABLE INSTITUTO. Alumnos (

DNI_A CHAR (9) UNIQUE PRIMARY KEY,

Nombre VARCHAR (25)

Apellido VARCHAR (45),

Edad TINYINT (80));

ENGINE=InnoDB;

- Tabla Matriculas, correspondiente a la entidad matricula con los atributos [(Curso_escolar, nota) y las claves Foráneas (Cod_Asig, DNI_A)] ;

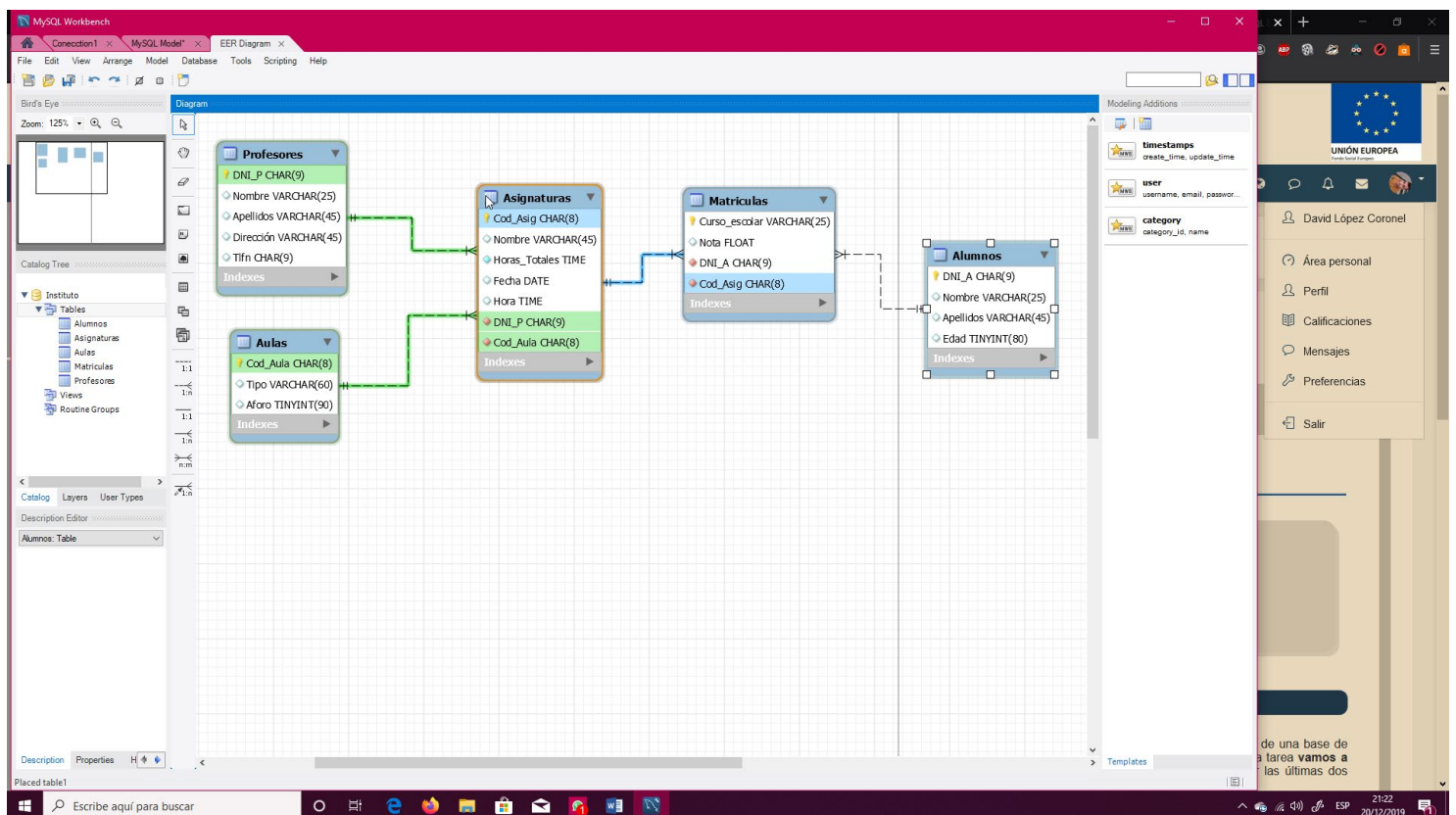
CREATE TABLE INSTITUTO. Matriculas (

Nota FLOAT (5.2),

Curso_escolar VARCHAR (25)

CONSTRAINT ASIGNATURAS, ALUMNOS_F_FK FOREIGN KEY (Cod_Asig, DNI_A)

REFERENCES ASIGNATURAS, ALUMNOS (Cod_Asig, DNI_A) ON DELETE SET NULL);



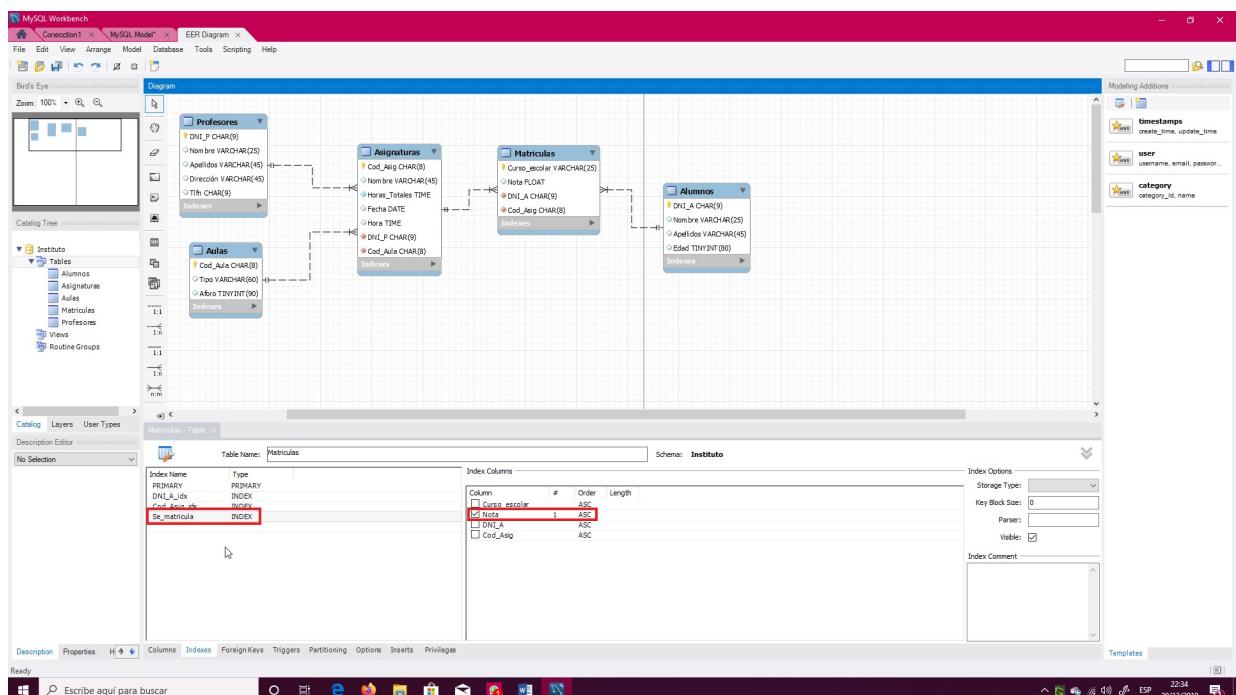
B) Después de crear las tablas, vamos a realizar algunas modificaciones sobre ellas. Escribe las sentencias SQL apropiadas para cada uno de los subapartados siguientes:

1. Eliminar el campo edad de la tabla ALUMNO.

ALTER TABLE Alumnos DROP COLUMN (Edad);

2. Crear un índice en el campo nota de la tabla Se_matricula (Matriculas).

CREATE INDEX Se_matricula ON Matriculas (Nota);

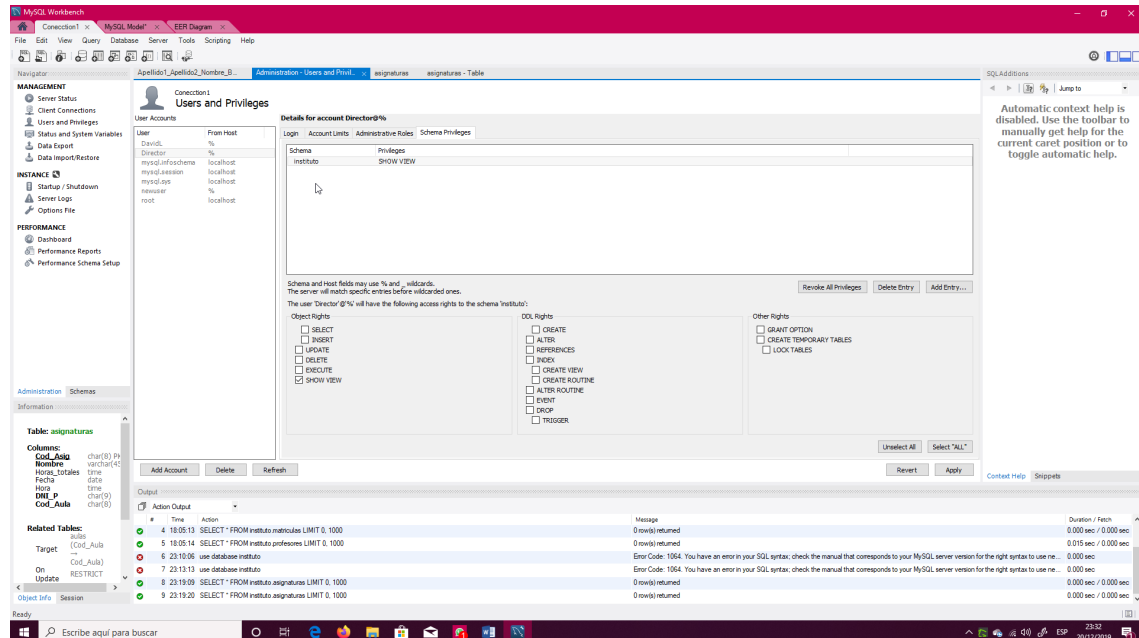


3. Añadir el campo estudios en la tabla PROFESOR y que éste sólo pueda tomar los valores "Diplomado", "Licenciado" o "Catedrático".

SELECT Diplomado, Licenciado, Catedrático FROM Profesores WHERE Estudios Is ('Diplomado', 'Licenciado', 'Catedrático');

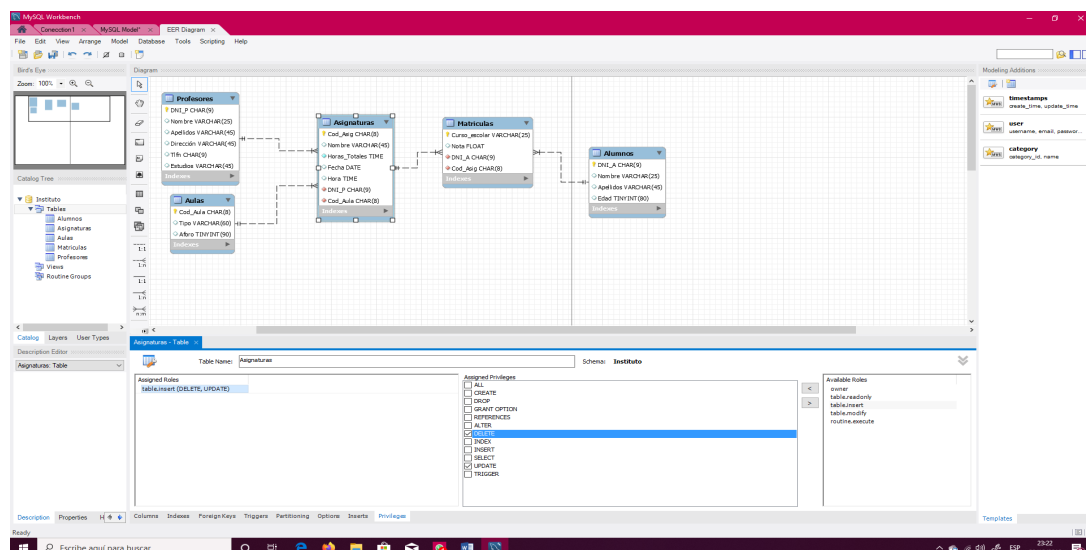
4. Crear el usuario "director" con la contraseña "admin" para que se pueda conectar desde cualquier equipo.

CREATE USER director IDENTIFIED BY 'admin';



5. Conceder los privilegios de actualizar y borrar datos al usuario "secretaria" en la tabla ASIGNATURAS.

GRANT UPDATE, DELETE ON Instituto.Asignaturas TO 'secretaria'@'localhost';



6. Quitar el privilegio para eliminar datos en la tabla ASIGNATURA al usuario "secretaria".

```
REVOKE DELETE ON Instituto. Asignaturas FROM 'secretaria'@'localhost';
```

David López Coronel