

### Actividad de aprendizaje 3-04

#### Se trata de:

- Modificar la estructura de varias tablas de la base de datos **centro** ya creadas en las actividades 3-01 y 3-02.

#### Objetivos:

- Crear tablas
- Añadir columnas a tablas existentes.
- Modificar y eliminar columnas en tablas existentes.
- Añadir índices en tablas existentes.
- Eliminar índices en tablas existentes.
- Añadir claves ajenas en tablas existentes.
- Eliminar claves ajenas en tablas existentes.

#### Procedimiento

Debes escribir las instrucciones SQL necesarias para realizar cada una de las siguientes tareas:

1.- Añade a la tabla **ALUMNOS** una columna **localidad** con un máximo de 20 caracteres. En la estructura de la tabla, esta columna debe aparecer después de la columna **direccion**.

**ALTER TABLE alumnos ADD localidad VARCHAR(20) AFTER direccion;**

2.- Establece que la columna **localidades** añadida anteriormente es un índice normal. El nombre del índice debe ser **k\_loc\_alumnos**.

**ALTER TABLE alumnos ADD INDEX k\_loc\_alumnos (localidad);**

3.- Establece que la columna **apellidos** de los alumnos es índice normal. Da un nombre de índice conforme al formato que se ha dado para el índice creado anteriormente.

**ALTER TABLE alumnos ADD INDEX k\_apellidos\_alumnos (apellidos);**

4.- Añade en la tabla alumnos una columna **NIE** (número de identificación escolar) con tipo de dato **INT**.

**ALTER TABLE alumnos ADD nie INT;**

5.- Establece que el NIE de **ALUMNOS** no se pueda repetir y que su nombre de índice sea **uk\_NIE\_alumnos**.

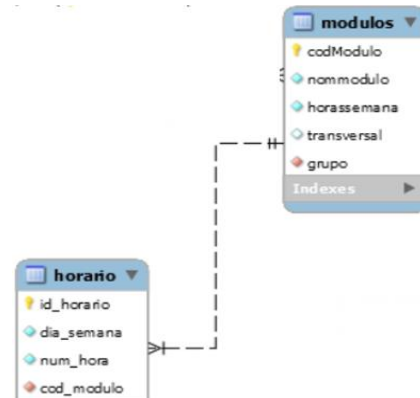
**ALTER TABLE alumnos ADD UNIQUE uk\_NIE\_alumnos (nie);**

6.- Nos hemos dado cuenta que no queríamos que NIE fuera INT. Queremos que sea un CHAR(6) dado que, aunque se representa con 6 dígitos, no es un dato objeto de realizar cálculos o comparaciones ser mayor o ser menor que. Modifica el tipo de dato de la columna. La columna sigue sin permitir valores repetidos.

**ALTER TABLE alumnos MODIFY nie CHAR(6);**

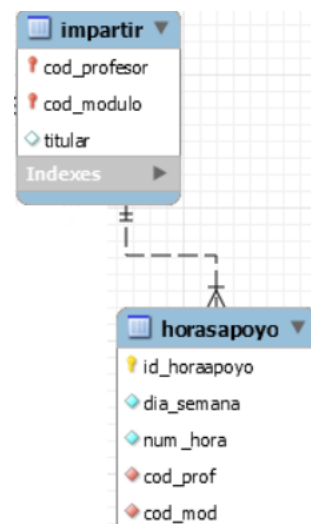
7.- Crea la tabla **HORARIO** de acuerdo como se representa en el diagrama relacional de la actividad de aprendizaje 3-01.

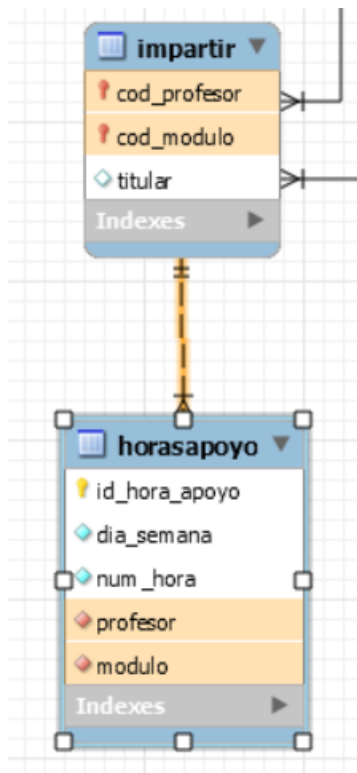
```
CREATE TABLE horario (  
  id_horario INT NOT NULL  
  AUTO_INCREMENT,  
  dia_semana TINYINT NOT NULL,  
  num_hora TINYINT NOT NULL,  
  cod_modulo CHAR(6) NOT NULL,  
  PRIMARY KEY (id_horario),  
  CONSTRAINT fk_modulo_horario FOREIGN KEY (cod_modulo)  
    REFERENCES modulos(codmodulo));
```



8.- Nos hemos dado cuenta de que el diagrama relacional para la tabla **HORASAPOYO** no es correcto. La clave ajena está relacionada con la tabla **PROFESOR** y debería estar relacionada con la tabla **IMPARTIR** para que se pueda saber el profesor que hace el apoyo y el módulo en que lo hace. Elimina la clave ajena de la tabla y añade una columna para el código del módulo y una relación de clave ajena entre **HORASAPOYO** e **IMPARTIR**.

```
CREATE TABLE horasapoyo (  
  id_horaapoyo INT NOT NULL AUTO_INCREMENT,  
  dia_semana TINYINT NOT NULL,  
  num_hora TINYINT NOT NULL,  
  cod_prof INT UNSIGNED NOT NULL,  
  cod_mod CHAR(6) NOT NULL,  
  PRIMARY KEY(id_horaapoyo),  
  UNIQUE uk_hora_horasapoyo (dia_semana,num_hora,cod_prof),  
  CONSTRAINT fk_profesor_horasapoyo FOREIGN KEY (cod_prof,cod_mod)  
    REFERENCES impartir(cod_profesor,cod_modulo) ON UPDATE CASCADE  
    ON DELETE CASCADE);
```





**ALTER TABLE horasapoyo DROP FOREIGN KEY fk\_profesor\_horasapoyo;**

**ALTER TABLE horasapoyo CHANGE cod\_prof profesor INT NOT NULL;**

**ALTER TABLE horasapoyo ADD modulo CHAR(6) NOT NULL;**