



# Tarea para BD02. Actividades Bloque I - Soluciones

## EJERCICIO 1:

*Vamos a crear las tablas para una Academia donde se imparten distintos cursos de informática.*

*Empezaremos creando con SQL las siguientes tablas:*

1. Tabla **ALUMNOS** recogerá información sobre el alumnado: Nombre, Apellido1, Apellido2, NIF, Dirección, Sexo, Fecha de Nacimiento y Curso en el que se matricula.
2. Tabla **CURSOS** con los siguientes campos: Nombre del Curso, Código del Curso que lo identifica, NIF del Profesor, Máximo número de alumnos/as recomendado, Fecha de inicio, Fecha final, Número de horas totales del curso. Los alumnos/as no pueden compaginar varios cursos a la vez.
3. Tabla **PROFESORES** con los siguientes campos: Nombre, Apellido1, Apellido2, NIF, Dirección, Titulación, Salario.
  - a. Debes elegir los nombres más adecuados para los atributos teniendo en cuenta las reglas.
  - b. Debes elegir los tipos de datos adecuados en función del contenido de los campos.
  - c. Debes establecer las siguientes restricciones:
    1. El alumno/a debe matricularse en un curso antes de que se le pueda dar de alta.  
Significa que curso será clave ajena en esta tabla y es el mismo campo que Código del curso de la tabla **CURSOS**.
    2. En un curso, el número de horas es un dato que no puede faltar, es obligatorio que contenga información.  
Significa que hay que poner una restricción NOT NULL.
    3. En la tabla **PROFESORES**, el atributo Salario no puede estar vacío.  
Significa que Salario es NOT NULL.
    4. Dos cursos no pueden llamarse de la misma forma.  
Significa que el nombre del curso será UNIQUE.



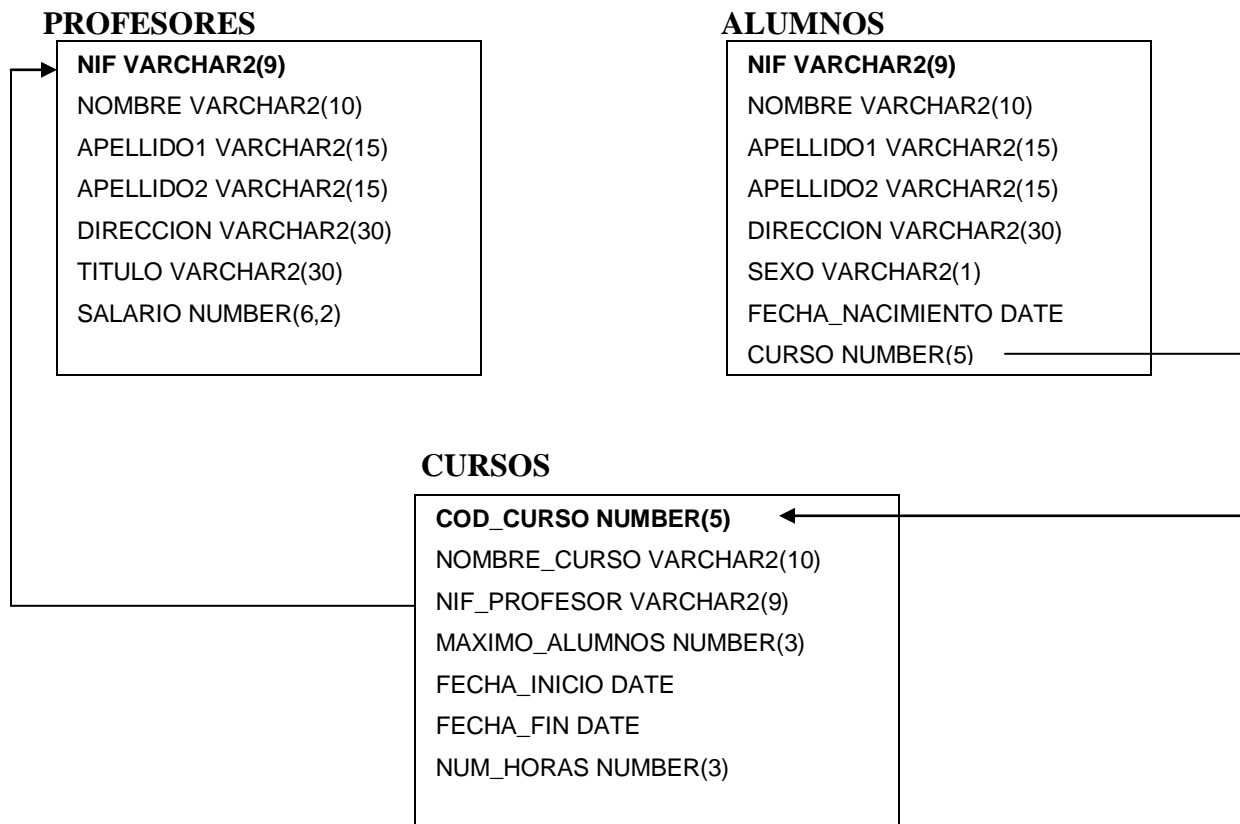
5. *Dos profesores no pueden llamarse igual.*  
Significa que el nombre del profesor será UNIQUE.
6. *Podremos diferenciar las tuplas de la tabla CURSOS por el Código del Curso.*  
Luego el Código del Curso será clave primaria.
7. *Podremos diferenciar las tuplas de la tabla PROFESORES y ALUMNOS por el NIF.*  
Por tanto, NIF será clave primaria en esas tablas.
8. *La fecha de comienzo del curso nunca puede ser mayor que la fecha de finalización.*  
Significa que tenemos que poner una restricción de validación
9. *El dominio del atributo sexo es M (mujer) y H (hombre).*  
Pondremos una restricción para que únicamente pueda tomar estos dos valores.
10. *Se debe cumplir la regla de integridad referencial de borrado en cascada.*  
Para que se cumpla la integridad referencial podemos poner en las restricciones de claves ajenas ON DELETE CASCADE después de REFERENCES.

Las tablas nos quedarían así

```
CREATE TABLE PROFESORES (  
    NOMBRE VARCHAR2(10) UNIQUE,  
    APELLIDO1 VARCHAR2(15),  
    APELLIDO2 VARCHAR2(15),  
    NIF VARCHAR2(9),  
    DIRECCION VARCHAR2(30),  
    TITULO VARCHAR2(30),  
    SALARIO NUMBER(6,2) NOT NULL,  
    CONSTRAINT profesores_nif_PK PRIMARY KEY(NIF)  
);  
  
CREATE TABLE CURSOS (  
    NOMBRE_CURSO VARCHAR2(10) UNIQUE,  
    COD_CURSO NUMBER(5) PRIMARY KEY,  
    NIF_PROFESOR VARCHAR2(9),  
    MAXIMO_ALUMNOS NUMBER(3),  
    FECHA_INICIO DATE,  
    FECHA_FIN DATE,  
    NUM_HORAS NUMBER(3) NOT NULL,  
    CONSTRAINT cur_nif_FK FOREIGN KEY (NIF_PROFESOR)  
        REFERENCES PROFESORES (NIF) ON DELETE CASCADE,  
    CONSTRAINT cursos_fec_CK CHECK(FECHA_FIN>FECHA_INICIO)  
);
```



```
CREATE TABLE ALUMNOS (  
    NOMBRE VARCHAR2(10),  
    APELLIDO1 VARCHAR2(15),  
    APELLIDO2 VARCHAR2(15),  
    NIF VARCHAR2(9) PRIMARY KEY,  
    DIRECCION VARCHAR2(30),  
    SEXO VARCHAR2(1),  
    FECHA_NACIMIENTO DATE,  
    CURSO NUMBER(5),  
    CONSTRAINT alumnos_cur_FK FOREIGN KEY (CURSO)  
        REFERENCES CURSOS (COD_CURSO),  
    CONSTRAINT alumnos_sex_CK CHECK (SEXO IN ('H','M'))  
);
```



## EJERCICIO2:

*Vamos a modificar las tablas que hemos creado en el apartado anterior:*

1. *Crea un nuevo atributo llamado Edad de tipo numérico a la tabla ALUMNOS.*

```
ALTER TABLE ALUMNOS ADD EDAD NUMBER(3);
```

*Añade las siguientes restricciones:*

2. *Modifica el campo que has creado anteriormente para que la edad del alumno/a esté comprendida entre 14 y 65 años.*

```
ALTER TABLE ALUMNOS ADD  
CONSTRAINT alumnos_edad_CK CHECK(EDAD BETWEEN 14 AND 65);
```



3. *Modifica el campo Número de horas del CURSO de manera que solo pueda haber cursos con 30, 40 o 60 horas.*

```
ALTER TABLE CURSOS ADD  
CONSTRAINT cursos_num_horas_CK CHECK (NUM_HORAS IN (30, 40, 60));
```

4. *No podemos añadir un curso si su número máximo de alumnos es inferior a 15.*


```
ALTER TABLE CURSOS ADD CONSTRAINT cursos_alu_max_CK CHECK(MAXIMO_ALUMNOS >=15);
```

5. *Elimina la restricción que controla los valores que puede tomar el atributo Sexo.*

```
ALTER TABLE ALUMNOS DROP CONSTRAINT alumnos_sex_CK;
```

6. *Elimina la columna Dirección de la tabla PROFESORES.*

```
ALTER TABLE PROFESORES DROP COLUMN DIRECCION;
```

7. *Cambia la clave primaria de la tabla PROFESORES por Nombre y Apellidos. Insertar una fila con información a la tabla profesores. Si se hace desde el SQLDeveloper pulsar al final el icono  para que la inserción en la tabla sea definitiva.*

Como hay una clave ajena desde CURSOS a PROFESORES asociada al NIF deberemos eliminar esta restricción antes de nada para que nos deje eliminar la restricción de clave primaria en la tabla profesores

```
ALTER TABLE CURSOS DROP CONSTRAINT cursos_nif_Fk;
```

Borramos la clave primaria

```
ALTER TABLE PROFESORES DROP CONSTRAINT profesores_nif_PK CASCADE;
```

Ahora creamos la nueva clave primaria

```
ALTER TABLE PROFESORES ADD CONSTRAINT profesores_apef_PK PRIMARY KEY(NOMBRE, APELLIDO1,  
APELLIDO2);
```

Insertamos una fila con información en dicha tabla usando el asistente

8. *Renombra la tabla PROFESORES por TUTORES.*

```
RENAME PROFESORES TO TUTORES;
```

9. *Elimina la tabla ALUMNOS*

```
DROP TABLE ALUMNOS;
```



10. Crea un usuario con el nombre *BD02* y clave *BD02* y dale todos los privilegios sobre la tabla

*TUTORES*.

```
CREATE USER BD02 IDENTIFIED BY BD02;  
GRANT ALL ON TUTORES TO BD02;
```

11. Comprobar si es posible el usuario *BD02* ver dicha tabla

Para ver una tabla desde un esquema ***select \* from usuario.nom\_tabla*** o desde nuestro esquema acceder al esquema del usuario y pulsar sobre su objeto tablas

12. Ahora al usuario anterior quítale permisos para modificar o actualizar la tabla *TUTORES*.

```
REVOKE ALTER, UPDATE ON TUTORES FROM NOMBREDELALUMNO;
```

13. Comprobar si es posible el usuario *BD02* ver dicha tabla ahora.