

Tarea para BD02.

Fecha tope de entrega: 24 de Noviembre de 2019

Detalles de la tarea de esta unidad.

Enunciado.

IMPORTANTE: Antes de comenzar debes crear un documento en el que tendrás que anotar las respuestas de los ejercicios.

EJERCICIO 1:

Vamos a crear las tablas para una Academia donde se imparten distintos cursos de informática. Empezaremos creando con SQL las siguientes tablas:

Tabla ALUMNOS recogerá información sobre el alumnado: Nombre, Apellido1, Apellido2, NIF, Dirección, Sexo, Fecha de Nacimiento y Curso en el que se matricula.

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.

Tabla PROFESORES con los siguientes campos: Nombre, Apellido1, Apellido2, NIF, Dirección, Titulación, Salario.

Debes elegir los nombres más adecuados para los atributos teniendo en cuenta las reglas.

Debes elegir los tipos de datos adecuados en función del contenido de los campos.

Debes establecer las siguientes restricciones:

El alumno o alumna debe matricularse en un curso antes de que se le pueda dar de alta.

En un curso, el número de horas es un dato que no puede faltar, es obligatorio que contenga información.

En la tabla PROFESORES, el atributo Salario no puede estar vacío.

Dos cursos no pueden llamarse de la misma forma.

Dos profesores no pueden llamarse igual.

Podremos diferenciar las tuplas de la tabla CURSOS por el Código del Curso.

Podremos diferenciar las tuplas de la tabla PROFESORES y ALUMNOS por el NIF.

La fecha de comienzo del curso nunca puede ser mayor que la fecha de finalización.

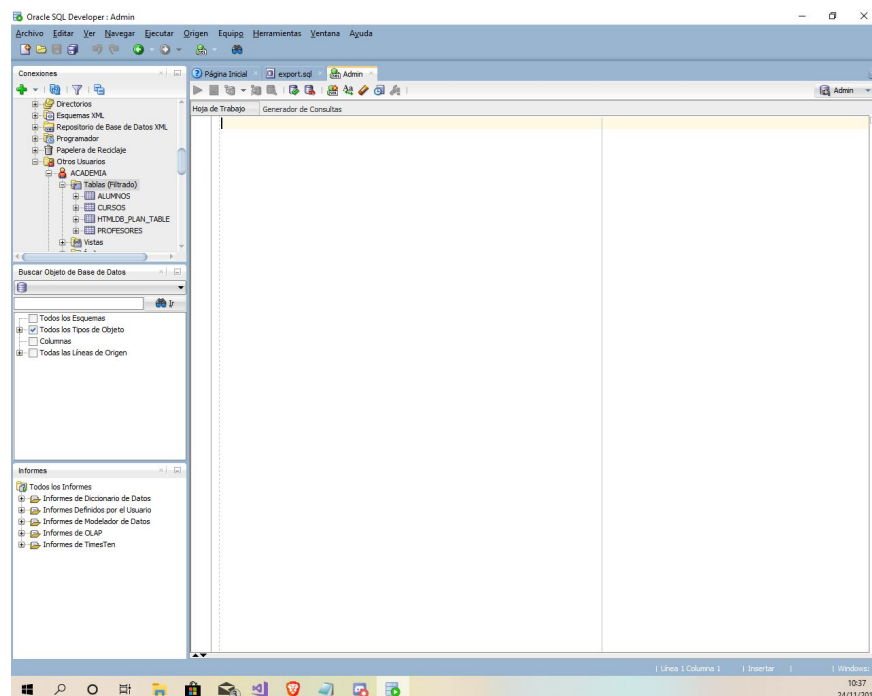
El dominio del atributo sexo es M (mujer) y H (hombre).

Se debe cumplir la regla de integridad referencial.

SOLUCIÓN PROPUESTA:

Para la elaboración de la tarea, estoy utilizando Oracle SQL Developer, he creado un usuario llamado Academia, estoy logueado con las credenciales y a partir de aquí estoy trabajando con él, por ese motivo cuando creo la tabla, hago uso de la nomenclatura `CREATE TABLE "ACADEMIA"."ALUMNOS"`, hago referencia al usuario primero y luego a la tabla.

TABLAS CREADAS:



1- TABLA ALUMNOS:

```
CREATE TABLE "ACADEMIA"."ALUMNOS" (  
    "NIF_ALUMNO" VARCHAR2(9),  
    "NOMBRE_ALUMNO" VARCHAR2(30),  
    "APELLIDO1_ALUMNO" VARCHAR2(15),  
    "APELLIDO2_ALUMNO" VARCHAR2(15),  
    "DIRECCION_ALUMNO" VARCHAR2(150),  
    "SEXO_ALUMNO" VARCHAR2(1),  
    "FECHA_NAC_ALUMNO" DATE,  
    "CURSO_MATRICULA_ALUMNO" VARCHAR2(15) NOT NULL ENABLE,  
    CONSTRAINT "ALUM_NIF_PK" PRIMARY KEY ("NIF_ALUMNO");
```

2- TABLA CURSOS:

```
CREATE TABLE "ACADEMIA"."CURSOS" (  
    "COD_CURSO" VARCHAR2(10) NOT NULL ENABLE,  
    "NOMBRE_CURSO" VARCHAR2(15),  
    "NIF_PROFESOR" VARCHAR2(9),  
    "NUM_MAX_ALUMNOS" NUMBER,  
    "FECHA_INICIO" DATE,  
    "FECHA_FIN" DATE,  
    "NUM_HORAS_CURSO" NUMBER NOT NULL ENABLE,  
    CONSTRAINT "COD_CUR_PK" PRIMARY KEY ("COD_CURSO");
```

3- TABLA PROFESORES:

```
CREATE TABLE "ACADEMIA"."PROFESORES" (  
    "NIF_PROF" VARCHAR2(9) NOT NULL ENABLE,  
    "NOMBRE_PROF" VARCHAR2(20),  
    "APELLIDO1_PROF" VARCHAR2(15),  
    "APELLIDO2_PROF" VARCHAR2(15),  
    "DIRECCION_PROF" VARCHAR2(150),  
    "TITULO_PROF" VARCHAR2(20 BYTE),  
    "SALARIO_PROF" NUMBER NOT NULL ENABLE,  
    CONSTRAINT "PROF_NIF_PK" PRIMARY KEY ("NIF_PROF");
```

Se ha dado respuesta a todas las preguntas relacionada a esta unidad 1, en la ejecución de primera tarea, se ha puesto en práctica las siguientes restricciones:

Para la tabla ALUMNOS:

El dominio del atributo sexo es M (mujer) y H (hombre):

```
CONSTRAINT "ALUM_SEX_CK" CHECK (SEXO_ALUMNO IN ('M', 'H')) ENABLE
```

```
CONSTRAINT "ALUM_COD_PK" FOREIGN KEY ("NIF_ALUMNO")  
REFERENCES "ACADEMIA"."CURSOS" ("COD_CURSO") ENABLE
```

Para la tabla CURSOS:

```
CONSTRAINT "FEC_CUR_CK" CHECK (FECHA_INICIO < FECHA_FIN) ENABLE,  
CONSTRAINT "PRO_NIF_FK" FOREIGN KEY ("COD_CURSO")  
REFERENCES "ACADEMIA"."PROFESORES" ("NIF_PROF") ENABLE
```

EJERCICIO2:

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

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

Añade las siguientes restricciones:

Creemos el nuevo campo:

```
ALTER TABLE "ACADEMIA"."ALUMNOS" ADD(  
                                EDAD NUMBER(2)  
                                );
```

Modifica el campo que has creado anteriormente para que la edad del alumno o alumna esté comprendida entre 14 y 65 años.

```
ALTER TABLE "ACADEMIA"."ALUMNOS" MODIFY(  
EDAD NUMBER(2) CONSTRAINT EDAD_ALUM_CK CHECK (EDAD BETWEEN 14 AND 65)  
);
```

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

```
ALTER TABLE "ACADEMIA"."CURSOS" ADD (  
CONSTRAINT HOR_NUM_CK CHECK(NUM_HORAS_CURSO IN(30, 40, 60))  
);
```

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

```
ALTER TABLE "ACADEMIA"."CURSOS" ADD (  
CONSTRAINT NUM_CUR_CK CHECK(NUM_MAX_ALUMNOS > 15)  
);
```

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

```
ALTER TABLE "ACADEMIA"."ALUMNOS" DROP CONSTRAINT ALUM_SEX_CK;
```

Elimina la columna Dirección de la tabla PROFESORES.

```
ALTER TABLE "ACADEMIA"."PROFESORES" DROP COLUMN DIRECCION_PROF;
```

Cambia la clave primaria de la tabla PROFESORES por Nombre y Apellidos.

PASO1:

Borramos clave primaria de profesores:

```
ALTER TABLE "ACADEMIA"."PROFESORES" DROP CONSTRAINT PROF_NIF_PK;
```

PASO2:

```
ALTER TABLE "ACADEMIA"."PROFESORES" ADD CONSTRAINT PROF_NA1A2_PK  
PRIMARY KEY(NOMBRE_PROF, APELLIDO1_PROF, APELLIDO2_PROF);
```

Renombra la tabla PROFESORES por TUTORES.

```
ALTER TABLE "ACADEMIA"."PROFESORES" RENAME TO TUTORES;
```

Elimina la tabla ALUMNOS.

```
DROP TABLE "ACADEMIA"."ALUMNOS" CASCADE CONSTRAINT;
```

Crea un usuario con tu nombre y clave BD02 y dale todos los privilegios sobre la tabla CURSOS.

PASO1: CREAMOS EL USUARIO.

```
CREATE USER JUAN IDENTIFIED BY BD02;
```

PASO2: DAMOS LOS PERMISOS Y PRIVILEGIOS.

```
GRANT ALL ON "ACADEMIA"."CURSOS" TO JUAN;
```

Ahora al usuario anterior quítale permisos para modificar o actualizar la tabla CURSOS.

```
REVOKE ALTER, UPDATE ON "ACADEMIA"."CURSOS" FROM JUAN;
```