

Ejercicio de Base de Datos, DDL y DML.



-- Base de Datos Academia

-- 1-. Crear la Base de Datos.

-- 2-. Insertando los Datos en la Tablas, Hay que Insertarlos en ese Orden Debido a las Restricciones de las FOREIGN KEY. Para Crear la Tabla cursos Debe Existir la Tabla Profesores y Para Crear la Tabla alumnos Debe Existir la Tabla cursos.

-- 3-. Inserta el Alumno:

-- 4-. Agrega el Campo edad a la Tabla profesores.

-- 5-. Agregando Algunas Restricciones.

-- 6-. Eliminando la Restricción Para el Atributo sexo.

-- 7-. Modificar la Columna curso de la Tabla alumnos para que sea UNIQUE.

-- 8-. Eliminando la Restricción NOT NULL del Atributo gana de la Tabla de profesores.

-- 9-. Agregando la Restricción NOT NULL al Atributo fecha_inicio de la Tabla cursos.

-- 10-. Cambiando la Clave Primaria(PRIMARY KEY) de la Tabla profesores a Nombre, Apellido1, Apellido2.

-- 11-. Insertando Tuplas en la Tabla profesores y en la Tabla alumnos.

-- 12-. Modificando la Fecha de Nacimiento de Antonia López a 23 de Diciembre de 1986.

-- 13-. Intentando Cambiar a la Alumna Antonia López al Curso 5.

-- 14-. Eliminando de la Tabla profesores a la Profesora Laura Jiménez.

-- 15-. Creando una Tabla de Uso Temporal nombre de alumnos, con el Atributo Nombre Completo de Tipo Cadena de Caracteres y con el Contenido de la Tabla alumnos en Esos Campos, Sin PRIMARY KEY.

-- 16-. Borrando Todas las Tablas.

-- Base de Datos Academia: El Documento Original está Hecho en Formato SQL, las Líneas que Comienzan por -- Son Comentarios.

-- 1-. Creo la Base de Datos.

```
CREATE DATABASE academia;
```

-- La Uso.

```
USE academia;
```

-- Creo la Tabla alumnos.

```
CREATE TABLE alumnos (  
    Nombre varchar(20) NOT NULL,  
    Apellido1 varchar(20) NOT NULL,  
    Apellido2 varchar(20),  
    dni varchar(9),  
    Direccion varchar(128) NOT NULL,  
    sexo char,  
    fecha_nacimiento date,  
    curso int NOT NULL  
);
```

-- Creo la Tabla cursos.

```
CREATE TABLE cursos (  
    cod_curso int,
```

```
Nombre_curso varchar(32) NOT NULL,  
dni_profesor varchar(9),  
maximo_alumnos tinyint(4),  
fecha_inicio date NOT NULL,  
fecha_fin date NOT NULL,  
num_horas int NOT NULL  
);
```

-- Creo la Tabla profesores.

```
CREATE TABLE profesores (  
    Nombre varchar(20) NOT NULL,  
    Apellido1 varchar(20) NOT NULL,  
    Apellido2 varchar(20),  
    dni varchar(9),  
    Direccion varchar(128) NOT NULL,  
    titulo varchar(48) NOT NULL,  
    gana decimal(5,2) NOT NULL  
);
```

```
ALTER TABLE cursos  
    MODIFY cod_curso int AUTO_INCREMENT PRIMARY KEY,  
    ADD UNIQUE KEY Nombre_curso (Nombre_curso),  
    ADD KEY `Hace Referencia a dni de la Tabla profesores`  
    (dni_profesor);
```

```
ALTER TABLE alumnos  
    ADD PRIMARY KEY (dni),  
    ADD KEY `Hace Referencia a cod_curso de la Tabla cursos` (curso);
```

```
ALTER TABLE profesores  
    ADD PRIMARY KEY (dni);
```

```
ALTER TABLE alumnos
```

```
    ADD CONSTRAINT sexo CHECK (BINARY sexo = UPPER('H')  
OR BINARY sexo = UPPER('M')),
```

```
    ADD CONSTRAINT FK_cod_curso FOREIGN KEY (curso)  
REFERENCES cursos (cod_curso) ON DELETE CASCADE ON  
UPDATE CASCADE;
```

```
ALTER TABLE cursos
```

```
    ADD CONSTRAINT fechas CHECK (fecha_inicio < fecha_fin),  
    ADD CONSTRAINT FK_dni_profesor FOREIGN KEY (dni_profesor)  
REFERENCES profesores (dni) ON DELETE NO ACTION ON UPDATE  
CASCADE;
```

-- 2-. Insertando los Datos en las Tablas, Hay que Insertarlos en ese Orden Debido a las Restricciones de las FOREIGN KEY, Para Insertar en la Tabla cursos Debe Haber Datos en la Tabla Profesores y Para Insertar en la Tabla alumnos Debe Haber Datos en la Tabla cursos.

```
INSERT INTO profesores VALUES ("Juan", "Arch", "López", "32432455",  
"Puerta Negra 4", "Ing. Informática", 80.25), ("María ", "Oliva ", "Rubio ",  
"43215643 ", "Juan Alfonso 32", "Lda. Fil. Inglesa", 75.25);
```

```
INSERT INTO cursos VALUES (1, "Inglés Básico", "43215643", 15,  
"2000-11-01", "2000-12-22", 120), (2, "Administración Linux",  
"32432455", NULL, "2000-09-01", NULL, 80);
```

-- Al Intentar Insertar el Curso N° 2 sin Fecha de Finalización da Error Debido a la Restricción de que la Fecha de Inicio debe ser Anterior a la Fecha de Fin. Agregando una Fecha Posterior en la columna fecha_fin, Como se Puede Ver a Continuación, se Puede Insertar.

```
INSERT INTO cursos VALUES (1, "Inglés Básico", "43215643", 15,  
"2000-11-01", "2000-12-22", 120), (2, "Administración Linux",  
"32432455", NULL, "2000-09-01", "2000-11-30", 80);
```

```
INSERT INTO alumnos VALUES ("Lucas", "Manilva", "López", "123523",  
"Alhamar 3", "H", "1979-11-01", 1), ("Antonia", "López", "Alcántara",  
"2567567", "Maniquí 21", "M", NULL, 2), ("Manuel", "Alcántara",  
"Pedrós", "3123689", "Julián 2", NULL, NULL, 2), ("José", "Pérez",  
"Caballar", "4896765", "Jarcha 5", "H", "1977-02-03", 1);
```

-- 3-. Inserta el Alumno:

```
INSERT INTO alumnos VALUES ("Sergio", "Navas", "Retal", "123523",  
NULL, "P", NULL, NULL);
```

-- El Primer Error que Sale es: #1048 - La columna 'Dirección' no puede ser nula, ya que al Crear la Tabla Declare la Columna Dirección Como NOT NULL.

-- Si Inserto una Dirección Sale el Error: #1048 - La columna 'curso' no puede ser nula, ya que la Columna curso la Creé con la Restricción NOT NULL.

-- Si Pongo uno de los Cursos de la Tabla cursos, Sale el Error: #4025 - No se cumple la RESTRICCIÓN `sexo` para `academia`.`alumnos`. Ya que se Puso una Restricción Para que Solo admita 'H', 'M' (Mayúsculas) o NULL.

-- Si Soluciono lo del Sexo Muestra el Error: #1062 - Entrada duplicada '123523' para la clave 'PRIMARY', El dni es Clave Primaria y ya Estaba Registrado para el Alumno Lucas Manilva López.

-- Si Pongo un dni Válido, se Inserta la Tupla en la Tabla alumnos.

-- 4-. Agrega el Campo edad a la Tabla profesores.

```
ALTER TABLE profesores ADD edad tinyint(4);
```

-- 5-. Agregando Algunas Restricciones.

```
ALTER TABLE profesores  
  ADD CONSTRAINT edad CHECK (edad >= 24 AND edad <= 62);
```

```
ALTER TABLE cursos  
  ADD CONSTRAINT max_alumnos CHECK (maximo_alumnos >= 10);
```

```
ALTER TABLE cursos  
  ADD CONSTRAINT num_horas CHECK (num_horas > 100);
```

-- Al Poner la Restricción del Número de Horas de los cursos Mayor que 100 da el Error: #4025 - No se cumple la RESTRICCIÓN `num_horas` para `academia`.`cursos`. Debido a que ya Existe un Curso que Tiene una Duración de 80 Horas, Cambio Provisionalmente la Duración a 120 Horas, Para Poder Aplicar la Restricción.

```
UPDATE cursos SET num_horas = 120 WHERE cod_curso = 2;
```

-- Al Volver a Aplicar la Restricción de que Número de Horas Tiene que ser Mayor que 100 Funciona Perfectamente.

-- 6-. Eliminando la Restricción Para el Atributo sexo.

```
ALTER TABLE alumnos  
  DROP CONSTRAINT sexo;
```

-- Con Este Comando DDL Elimino la Restricción para Usar Solo 'H' o 'M' en la Columna sexo de los Alumnos, al Ejecutarla Pide Confirmación.

-- 7-. Modificar la Columna curso de la Tabla alumnos para que sea UNIQUE.

```
ALTER TABLE alumnos  
  MODIFY COLUMN curso int UNIQUE;
```

-- Si Intento Modificar la Columna curso Sale el Error: #1062 - Entrada duplicada '1' para la clave 'curso'. Ya que Hay 4 Alumnos y 2 Están en el Curso 1 y los Otros 2 en el Curso 2.

-- 8-. Eliminando la Restricción NOT NULL del Atributo gana de la Tabla de profesores.

```
ALTER TABLE profesores  
  MODIFY COLUMN gana decimal(5,2);
```

-- Se Usa el Comando DDL Anterior Para Eliminar la Restricción NOT NULL del Atributo gana de la Tabla profesores, ya que MySQL Pone NULL por Defecto al Crear/Modificar una Columna.

-- 9-. Agregando la Restricción NOT NULL al Atributo fecha_inicio de la Tabla cursos.

```
ALTER TABLE cursos  
  MODIFY COLUMN fecha_inicio date NOT NULL;
```

-- Se Usa para Agregar la Restricción NOT NULL a la Columna fecha_inicio de la Tabla cursos.

-- 10-. Cambiando la Clave Primaria(PRIMARY KEY) de la Tabla profesores a Nombre, Apellido1, Apellido2.

```
ALTER TABLE cursos  
    DROP CONSTRAINT FK_dni_profesor;
```

```
ALTER TABLE profesores  
    DROP PRIMARY KEY,  
    ADD PRIMARY KEY (Nombre, Apellido1, Apellido2);
```

-- De Esta Manera se Puede Eliminar la PRIMARY KEY de una Tabla y Declarar Como PRIMARY KEY a otro Atributo o Conjunto de Atributos(Clave Primaria Compuesta) Como en Este Caso. Como la PRIMARY KEY dni Está Siendo Usada por una FOREIGN KEY de la Tabla cursos, Primero Hay que Eliminar la Relación de la Clave Foránea(Restricción FK_dni_profesor) y Después Proceder al Cambio de PRIMARY KEY.

-- 11-. Insertando Tuplas en la Tabla profesores y en la Tabla alumnos.

```
INSERT INTO profesores VALUES ("Juan", "Arch", "López", "32432455",  
"Puerta Negra 4", "Ing. Informática", NULL);
```

-- El Primer Error que Mostrará este Comando DML es: #1136 - El número de columnas no se corresponde con el número de valores en la línea 1, ya que se Agregó a la Tabla profesores el Campo edad. Si Agrego una Edad Dará el Siguiete Error: #1062 - Entrada duplicada

'Juan-Arch-López' para la clave 'PRIMARY'. Ya que la Clave Primaria Compuesta Nombre, Apellido1, Apellido2 ya Contiene los Datos que se Están Intentando Insertar.

```
INSERT INTO profesores VALUES ("Juan", "Arch", "López", "32432455",  
"Puerta Negra 4", "Ing. Informática", NULL, 30);
```

```
INSERT INTO alumnos VALUES ("María", "Jaén", "Sevilla", "789678",  
"Martos 5", "M", "1977-3-10", 3);
```

-- Este Comando DML da Error ya que la FOREIGN KEY FK_cod_curso Apunta a un Curso que no Existe(cod_curso = 3), el Error es: #1452 - No puedo añadir o actualizar una fila hija: falla una restricción de clave foránea (`academia`.`alumnos`, CONSTRAINT `FK_cod_curso` FOREIGN KEY (`curso`) REFERENCES `cursos` (`cod_curso`)). Si Cambio el Número de Curso a un Curso Válido se Inserta la Tupla Perfectamente.

```
INSERT INTO alumnos VALUES ("María", "Jaén", "Sevilla", "789678",  
"Martos 5", "M", "1977-3-10", 2);
```

```
INSERT INTO profesores VALUES ("César Osvaldo", "Matelat",  
"Borneo", "42268151Q", "Como E.T. en Mi Casa", "Técnico Superior  
DAW?", 10.10);
```

-- Este Comando DML Mostrará el Error: #1136 - El número de columnas no se corresponde con el número de valores en la línea 1. Ya que Falta el Atributo edad Agregado Después de Crear la Tabla. Si Modifico el Comando Agregando la Edad Se Inserta Perfectamente.

```
INSERT INTO profesores VALUES ("César Osvaldo", "Matelat",  
"Borneo", "42268151Q", "Como E.T. en Mi Casa", "Técnico Superior  
DAW?", 10.10, 55);
```

```
INSERT INTO alumnos VALUES ("César Osvaldo", "Matelat", "Borneo",  
"42268151Q", "Como E.T. en Mi Casa", "H", "1968-4-5", 2);
```

-- Este Comandos DML se Ejecuta Sin Errores.

-- 12-. Modificando la Fecha de Nacimiento de Antonia López a 23 de Diciembre de 1986.

```
UPDATE alumnos SET fecha_nacimiento = "1986-12-23" WHERE  
dni="2567567";
```

-- 13-. Intentando Cambiar a la Alumna Antonia López al Curso 5.

```
UPDATE alumnos SET curso = 5 WHERE dni="2567567";
```

-- Este Comando DML Sería Perfectamente Válido si no Fuera Porque no Existe el Curso Nº 5, por eso es que da el Error: #1452 - No puedo añadir o actualizar una fila hija: falla una restricción de clave foránea (`academia`.`alumnos`, CONSTRAINT `FK_cod_curso` FOREIGN KEY (`curso`) REFERENCES `cursos` (`cod_curso`)).

-- 14-. Eliminando de la Tabla profesores a la Profesora Laura Jiménez.

```
DELETE FROM profesores WHERE Nombre LIKE "Laura" AND  
Apellido1 LIKE "Jiménez";
```

-- En este Caso Podemos Intentar Eliminar un Campo de la Tabla profesores por el Nombre y el Primer Apellido de la Profesora, lo Ideal es Hacerlo por la Clave Primaria Completa que Para Esta Tabla es la Clave Combinada Nombre, Apellido1, Apellido2, Aunque lo Ideal es que

sea un Campo Más Único como el DNI por Ejemplo. Como en este Caso la Profesora Laura Jiménez no existe el Comando DML DELETE NO Dará Error, Pero Veremos que no ha Eliminado Ninguna Fila: 0 filas eliminadas. (La consulta tardó 0,0011 segundos.).

-- 15-. Creando una Tabla de Uso Temporal nombre_de_alumnos, con el Atributo Nombre_Completo de Tipo Cadena de Caracteres y con el Contenido de la Tabla alumnos en Esos Campos, Sin PRIMARY KEY.

```
CREATE TEMPORARY TABLE
nombre_de_alumnos(Nombre_Completo varchar(60)); INSERT INTO
nombre_de_alumnos SELECT CONCAT(Nombre, " ", Apellido1, " ",
Apellido2) FROM alumnos;
```

/* Para Obtener el Resultado Esperado Hay que Crear la Tabla Primero con su Atributo y Hacer el INSERT al Mismo Tiempo de los Datos Concatenados(Nombre, Apellido1, Apellido2) de la Tabla alumnos en PHPmyadmin. En la Consola de MySQL se Hace así:

```
MariaDB [academia]> /* CREATE TEMPORARY TABLE
nombre_de_alumnos(Nombre_Completo varchar(60));
/* Query OK, 0 rows affected (0.005 sec) */
```

```
/* MariaDB [academia]> /* INSERT INTO nombre_de_alumnos SELECT CONCAT(Nombre,
" ", Apellido1, " ", Apellido2) FROM alumnos;
/* Query OK, 1 row affected (0.003 sec) */
/* Records: 1 Duplicates: 0 Warnings: 0 */
```

```
/* MariaDB [academia]> /* SELECT * FROM nombre_de_alumnos;
/* +-----+
| Nombre_Completo      |
+-----+
| Alumno-1 Primero Segundo |
+-----+
1 row in set (0.000 sec) */
```

-- 16-. Borrando Todas las Tablas.

DROP TABLE nombre_de_alumnos;

DROP TABLE profesores;

DROP TABLE alumnos;

DROP TABLE cursos;

-- Se Pueden Eliminar las Tablas en ese Orden y Hay que Eliminar la Tabla alumnos Antes que la Tabla cursos. Debido a las Restricciones de las FOREIGN KEY, no se puede Eliminar la Tabla cursos Antes ya que la PRIMARY KEY de la Tabla cursos (cod_curso) Está Relacionada con la FOREIGN KEY curso de la Tabla alumnos. Aclaración, En Este Caso, se Puede Eliminar la Tabla profesores Antes que la Tabla cursos, ya que se Cambió la PRIMARY KEY de la Tabla profesores, Antes era dni y Estaba Relacionada con la FOREIGN KEY dni_profesor de la Tabla cursos si Aún Estuviera así Habría que Borrar Primero alumnos, Luego cursos y por Último profesores.