

DDL_II

Hemos visto...

- CREATE TABLE
 - NombreColumna TipoDato OpcionesColumna
 - NombreColumna: [0-9,a-z,A-Z\$_]
 - TipoDato: Número, Alfanumérico, Fechas/Horas, Binarios
 - OpcionesColumna: NULL, NOT NULL, DEFAULT, AUTO_INCREMENT
- ÍNDICES: primary key, unique
- Charset/Collation

→ Falta cómo implementar referencias a otras tablas.

INTEGRIDAD REFERENCIAL

SINTAXIS (lo añadiremos en el CREATE TABLE después de la definición de columnas)

```
[CONSTRAINT <nombre_restricción>] FOREIGN KEY (columna1,columna2,...)  
REFERENCES <tabla> [(columna1, columna2,...)]
```

Hay que considerar:

- Sólo con el motor InnoDB.
- Hay que definir un índice para las columnas implicadas. En MySql, el sistema lo hace por nosotros.

Ejemplo

```
CREATE TABLE tablaClase (  
    Id_Clase tinyint unsigned AUTO_INCREMENT PRIMARY KEY,  
    Descripcion_Clase varchar (100) NOT NULL DEFAULT 'Falta descripción')
```

```
CREATE TABLE tablaAlumno (  
    DNI_Alumno char(9) PRIMARY KEY,  
    nombreAlumno varchar(20) NOT NULL DEFAULT 'Sin nombre',  
    Clase_Alumno tinyint unsigned DEFAULT 0,  
    FOREIGN KEY (Clase_Alumno) REFERENCES tablaClase(Id_Clase))
```

Problema Integridad Referencial 1

- Implementa el ejemplo anterior
- Inserta filas en ambas tablas
- Elimina alguna fila en ambas tablas utilizando la siguiente estructura:

DELETE FROM <nombre_tabla> WHERE
<condición_que_debe_cumplir_la_fila_que_queremos_eliminar>

- Actualiza algún valor de algún campo en ambas tablas utilizando:

UPDATE <nombre_tabla> SET <columna>=<nuevo_valor> WHERE
<condición_que_debe_cumplir_la_fila_que_queremos_actualizar>

- Además, falta considerar cómo le indicamos al SGBDR cómo debe gestionar actualizaciones y modificaciones de columnas que referencian o son referencias por otras.
- FOREIGN KEY puede ir acompañada de dos expresiones:
 - ON DELETE <acción>: indica qué actuación automática queremos que MySql realice al eliminarse un registro de una columna referenciada por la columna cuyo comportamiento estamos definiendo.
 - ON UPDATE <acción>: Indica actuación en caso de actualización de la columna referenciada.
 - Las acciones posibles son:
 - NO ACTION o RESTRICT: Sinónimos. Acción por defecto. No permite eliminación/actualización en la tabla principal.
 - CASCADE: las eliminaciones/actualizaciones se propagan a las claves foráneas.
 - SET NULL: ante una eliminación/actualización de la columna padre, las hijas quedan a NULL.
 - SET DEFAULT: ante una eliminación/actualización de la columna padre, las hijas

Problema Integridad Referencial 2

- Haz SHOW CREATE TABLE para ver la estructura de las tablas del ejercicio anterior, ¿qué pasa con la clave foránea? ¿qué ha añadido el sistema?
- Visualizando los CREATE TABLE de las tablas de la base de datos 'world', ¿podrías escribir el modelo relacional de 'world'?
- Implementa el modelo relacional siguiente. Ten en consideración los siguientes puntos:
 - Ningún nombre puede ser nulo.
 - Todas las tablas deben incluir algún comentario explicando su contenido.
 - Los nombres de los concesionarios deben de ser únicos.
 - El campo Cifm de la tabla MARCA es autoincremental.
 - Los nombres pueden tener acentos y eñes.
 - No se debe permitir la eliminación de un registro apuntado por una o más claves foráneas.
 - Las actualizaciones de un registro apuntado por una más claves foráneas deben de propagarse en cascada.

NOTA: Supón los tipos de columna que consideres más adecuados en cada caso

- MARCA (Cifm, Nombre, Ciudad)
- COCHE (CodCoche, Nombre, Modelo, Cifm) DONDE {Cifm} REFIERE A MARCA
- CONCESIONARIO (Cifc, Nombre, Ciudad)
- CLIENTE(DNI, Nombre, Apellido, Ciudad)
- VENTA(Cifc, DNI, CodCoche, color) DONDE {Cifc} REFIERE A CONCESIONARIO, {CodCoche} REFIERE A COCHE, {DNI} REFIERE A CLIENTE

No olvides insertar alguna fila en cada tabla y realizar alguna consulta.