



GENERALITAT
VALENCIANA

ceedcv

CENTRE ESPECÍFIC
D'EDUCACIÓ A DISTÀNCIA DE
LA COMUNITAT VALENCIANA



UD.4: MODELO FÍSICO DDL

Prácticas no evaluables
Boletín B

Bases de Datos (BD)
CFGs DAM/DAW

Abelardo Martínez y Pau Miñana.
Basado y modificado de Sergio Badal y Raquel Torres.
Curso 2023-2024

Aspectos a tener en cuenta

Estas actividades son opcionales y no evaluables pero es recomendable hacerlas para un mejor aprendizaje de la asignatura.


⊘ Si buscas las soluciones por Internet o preguntas al oráculo de ChatGPT, te estarás engañando a ti mismo. Ten en cuenta que ChatGPT no es infalible ni todopoderoso. Es una gran herramienta para agilizar el trabajo una vez se domina una materia, pero usarlo como atajo en el momento de adquirir habilidades y conocimientos básicos perjudica gravemente tu aprendizaje.

Si lo utilizas para obtener soluciones o asesoramiento respecto a las tuyas, revisa cuidadosamente las soluciones propuestas igualmente. Intenta resolver las actividades utilizando los recursos que hemos visto y la documentación extendida que encontrarás en el "Aula Virtual".

ÍNDICE

- [1. Enunciados](#)
 - [1.1. Enunciado 1. Escalada](#)
 - [1.1.1. Crea el modelo físico](#)
 - [1.1.2. Script](#)
 - [1.1.3. Alteración de las tablas](#)
 - [1.2. Enunciado 2. Sucursales](#)
 - [1.3. Enunciado 3. Ventas](#)
- [2. Bibliografía](#)

1. Enunciados

 No uses NUNCA tildes, ni eñes, ni espacios, ni caracteres no alfanuméricos (salvo el guión bajo) en los metadatos (nombres de elementos de una base de datos) y usa letras minúsculas.

Realiza los ejercicios en MySQL. Crea una BD llamada *bdu4_b* que permita tildes en los datos.

1.1. Enunciado 1. Escalada

Traduce el siguiente Modelo Lógico Relacional a Modelo Físico en SQL.

Asociación (código, nombre, ubicación)

CP: código

Alpinistas (nif, nombre, f_nacimiento, ~~cod_aso~~, f_ingreso)

CP: nif

CAj: cod_aso → Asociación {código}

Teléfonos (nif, teléfono)

CP: {nif, teléfono}

CAj: nif → Alpinistas {nif}

Picos (nombre, altura, coordenadas, país)

CP: nombre

Escalada (nif, nombre, f_inicio, tiempo, oxígeno, cara)

CP: nif

CAj: nif → Alpinistas {nif}

CAj: nombre → Picos {nombre}

1.1.1. Crea el modelo físico

Traduce el Modelo Lógico Relacional anterior a Modelo Físico en MySQL.

1.1.2. Script

Escribe todas las sentencias de creación de las tablas en un archivo de texto plano con un editor de texto. Llama al script "alpinista.sql" y ejecútalo en MySQL.

1.1.3. Alteración de las tablas

Realiza las siguientes modificaciones sobre la base de datos una vez creadas las tablas:

- Añade un campo observaciones sobre la tabla Picos (decide tú el tipo y tamaño del mismo).
- Modifica el campo código de la tabla Asociación para que pueda contener números hasta el 999.999.
- Añade un campo foto sobre la tabla Alpinistas (decide tú el tipo y tamaño del mismo).
- Modifica la columna fecha_ingreso de la tabla Alpinistas para que por defecto tenga la fecha del sistema.
- El valor del campo coordenadas no debe repetirse nunca, a pesar de no ser clave primaria.
- Los Alpinistas deben tener al menos 18 años cuando se registran.
- Modifica la condición anterior para que no se puedan registrar hasta los 21 años.
- Si se borra un alpinista de la base de datos ¿qué debería hacerse con sus correspondientes entradas en la tabla Teléfonos? Indica la sentencia DDL necesaria para aplicar esta restricción.
- ¿Y si se borra una asociación de la base de datos?

1.2. Enunciado 2. Ventas

Traduce el siguiente Modelo Lógico Relacional y sus restricciones a Modelo Físico en MySQL.

Usuario (dni, nombre, f_nacimiento)

CP: dni

Cliente (dni, descuento)

CP: dni

CAj: dni → Usuario {dni}

Restricción CAj dni: un usuario no puede borrarse si es cliente.

Restricción Adicional:

descuento es un número entre 0'00 y 1'00.

Pedido (código, fecha, dni_cliente*)

CP: código

CAj: dni_cliente → Cliente {dni}

VNN: dni_cliente

Restricción CAj dni: un usuario no puede borrarse si tiene pedidos.

Línea (cod_ped, num_línea, cod_prod*, cantidad, precio_venta)

CP: {cod_ped, num_línea}

ÚNICO: {cod_ped, cod_prod}

CAj: cod_ped → Pedido {código} B:C M:C

CAj: cod_prod → Producto {cod_producto} B:R

VNN: cod_prod

Producto (cod_producto, nombre, precio_actual)

CP: cod_producto

1.3. Enunciado 3. Sucursales

Tenemos el siguiente script para la creación de unas tablas en Oracle. Adapta las instrucciones para que este script pueda ser ejecutado sobre una base de datos de MySQL.

Por otro lado, fíjate que no se ha creado ninguna clave ajena. Debes analizar cuáles son las claves ajenas necesarias para el correcto funcionamiento del sistema y añadirlas a posteriori a la base de datos por medio de modificaciones de tablas.

```
CREATE TABLE articulos (  
  id_art  INTEGER,  
  nombre  VARCHAR(30) DEFAULT '',  
  precio  NUMBER DEFAULT 0.0,  
  codigo  VARCHAR(7) DEFAULT '',  
  id_gru  INTEGER DEFAULT 0 NOT NULL,  
  CONSTRAINT art_id_pk PRIMARY KEY(id_art) );  
  
CREATE TABLE clientes (  
  id_cli  INTEGER NOT NULL ,  
  nombre  VARCHAR(60) DEFAULT '' NOT NULL ,  
  direccion  VARCHAR(80) DEFAULT '',  
  CONSTRAINT cli_idc_pk PRIMARY KEY (id_cli) );  
  
CREATE TABLE facturas (  
  id_fac  INTEGER NOT NULL ,  
  serie  VARCHAR(1) DEFAULT '' NOT NULL,  
  numero  INTEGER DEFAULT 0 NOT NULL,  
  fecha  DATE,  
  id_cli  INTEGER DEFAULT 0 NOT NULL,  
  id_ven  INTEGER NOT NULL,  
  CONSTRAINT fac_idf_pk PRIMARY KEY (id_FAC) );  
  
CREATE TABLE grupos (  
  id_gru  INTEGER,  
  descripcion  VARCHAR(15) DEFAULT '' NOT NULL,  
  CONSTRAINT gru_idg_pk PRIMARY KEY (id_gru) );  
  
CREATE TABLE lineas_factura (  
  id_lin  INTEGER NOT NULL,  
  id_fac  INTEGER DEFAULT 0 NOT NULL ,  
  importe  FLOAT DEFAULT 0 NOT NULL ,  
  id_art  INTEGER NOT NULL,  
  cantidad  INTEGER NOT NULL,  
  fecha  DATE DEFAULT '01/01/2000' NOT NULL ,  
  id_suc  INTEGER DEFAULT 0 NOT NULL ,  
  CONSTRAINT lin_linfac_pk PRIMARY KEY (id_lin, factura) );  
  
CREATE TABLE sucursales (  
  id_suc  INTEGER NOT NULL,  
  descripcion  VARCHAR(15) DEFAULT '' NOT NULL ,  
  CONSTRAINT suc_ids_pk PRIMARY KEY (id_suc) );  
  
CREATE TABLE vendedores (  
  id_ven  INTEGER NOT NULL,  
  nombre  VARCHAR(50) DEFAULT '' NOT NULL ,  
  fecha_ingreso  DATE DEFAULT '01-01-2000' NOT NULL ,  
  salario  FLOAT DEFAULT 0 NOT NULL ,  
  CONSTRAINT ven_idv_pk PRIMARY KEY (id_ven) );
```

2. Bibliografía

- MySQL 8.0 Reference Manual.
<https://dev.mysql.com/doc/refman/8.0/en/>
- Oracle Database Documentation
<https://docs.oracle.com/en/database/oracle/oracle-database/index.html>
- JavaTPoint. Difference between MySQL and Oracle.
<https://www.javatpoint.com/mysql-vs-oracle>
- W3Schools. MySQL Tutorial.
<https://www.w3schools.com/mysql/>
- GURU99. Tutorial de MySQL para principiantes Aprende en 7 días.
<https://guru99.es/sql/>
- Adam McGurk.How to change a foreign key constraint in MySQL
<https://dev.to/mcgurkadam/how-to-change-a-foreign-key-constraint-in-mysql-1cma>
- Sqlines. MySQL - SET FOREIGN_KEY_CHECKS.
http://www.sqlines.com/mysql/set_foreign_key_checks