



TEMA 4

EJERCICIOS (II)

Bases de Datos
CFGs DAW

Recopilacion de Ejercicios de:
Raquel Torres
Francisco Aldarias
Revisado por:
Pau Minana Climent
2020/2021

UD04. DISEÑO FÍSICO. DDL (II)

1. EJERCICIO 1

Traduce el siguiente modelo relacional al SQL de Oracle y MySQL:

Asociación (código, nombre, ubicación) CP:{código}

Alpinistas (nif, nombre, fecha_nacimiento, teléfonos, código, fecha_ingreso)

CP:{nif}

CA:{código} → Asociación

Mv (multivaluado): teléfonos

Teléfonos (nif, telefono)

CP:{nif,telefono}

CA:{nif} → Alpinistas

Picos (nombre, altura, coordenadas, país)

CP: {nombre}

Escala (nif, nombre, fecha_inicio, tiempo, oxígeno, cara)

CP{nif, nombre}

CA{nif} → alpinista

CA{nombre} → pico

1.1 Ejercicio 1.1

Antes de crear las tablas, vamos a crear el usuario alpinista, de manera que las tablas que vamos a crear se inserten en el esquema alpinista.

1.2 Ejercicio 1.2

Creamos las tablas, hazlo ejecutando el script alpinista.sql

1.3 Ejercicio 1.3

Añade un campo '*observaciones*' sobre la tabla PICOS (decide tú el tipo y tamaño del mismo).

1.4 Ejercicio 1.4

Modifica el campo *código* de la tabla **asociaciones** para que pueda contener números hasta el 999999.

1.5 Ejercicio 1.5

Añade un campo '*foto*' sobre la tabla **ALPINISTAS** (decide tú el tipo y tamaño del mismo).

1.6 Ejercicio 1.6

Modifica la columna *fecha_ingreso* de la table **alpinistas** para que por defecto tenga la fecha del sistema.

1.7 Ejercicio 1.7

El valor del campo '*coordenadas*' no debe repetirse nunca, a pesar de no ser clave primaria. Introduce la correspondiente restricción sobre **PICOS**.

1.8 Ejercicio 1.8

Si se borra un alpinista de la base de datos ¿qué debería hacerse con sus correspondientes entradas en la tabla **TELEFONOS**? Indica la sentencia DDL necesaria para aplicar esta restricción.

1.9 Ejercicio 1.9

¿Y si se borra una asociación de la base de datos?

1.10 Ejercicio 1.10

Desactivar la clave foránea de la tabla *alpinistas*.

1.11 Ejercicio 1.11

Volver a activar la clave foránea desactivada en la pregunta anterior.

2. EJERCICIO 2

Tenemos las siguientes tablas creadas, pero no se ha creado ninguna clave ajena. Se trata de que vosotros analicéis cuáles son las claves ajenas necesarias para el correcto funcionamiento del sistema y las añadáis a la base de datos por medio de modificaciones de las tablas (ALTER TABLE).

```
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 PRIMARYKEY(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_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)  
);
```

3. EJERCICIO 3

Traduce el siguiente modelo relacional al SQL de Oracle/MySQL y dibuja el diagrama del diseño físico

USUARIO (dni, nombre, fecha_nac)

CP: {DNI}

CLIENTE (dni, descuento)

CP: {dni}

CA: {dni} → Usuario

Restriccion CA dni : Un usuario no puede borrarse si es cliente.

PEDIDO (codigo, fecha, dni_cliente)

CP: {codigo}

CA: {dni_cliente} → Cliente

VNN: {dni_cliente}

Restriccion CA dni_cliente: Un cliente no puede borrarse si tiene lineas de pedido.

LINEA (numero_linea, precioventa, codigo_pedido, codigo_producto, cantidad)

CP : {numero_linea, codigo_pedido}

CA: {Codigo_pedido} → Pedido

CA: {codigo_producto}→Producto

VNN: {codigo_producto}

Restriccion CA Codigo_pedido: Borrado en cascada de todas las lineas del pedido

Restriccion CA Codigo_producto: Un producto no podra borrarse si existe en alguna linea de pedido.

PRODUCTO (codigo_producto, nombre, precioactual)

CP: codigo_producto

