

## **Bases de Datos: Ejercicios de SQL**

### **Conceptos Básicos de SQL -1**

**T5-EJ1** - ¿Para qué sirve el lenguaje de SQL y qué puedo hacer usándolo?

**T5-EJ2** - ¿Qué componentes tiene?

**T5-EJ3** - ¿Es estándar para todos los gestores de BD?

**T5-EJ4** - ¿Cómo defino una tabla y sus atributo?

**T5-EJ5** - ¿Qué tipos de datos pueden tener los atributos?

**T5-EJ6** - ¿Qué tipos de restricciones pueden tener los atributos?

**T5-EJ7** - ¿Cómo añado, modifico o elimino atributos o restricciones de una tabla?

**T5-EJ8** - ¿Cómo actualizo las filas de una tabla: insertar, borrar y actualizar?

**T5-EJ9** - ¿Qué es un índice? ¿Qué ventajas tiene? ¿Cómo se crea y se borra?

## Ejercicios sobre BDejemplo para implementar en ORACLE con SQL

### APARTADO 1: Creación de BDejemplo en Oracle.

**Se pide** crear en Oracle la base de datos BDejemplo, y cumpla las siguientes **restricciones adicionales**:

**Todas las tablas:**

- Las claves primarias deben tener algún valor.
- el DNI debe tener como 8 caracteres (no es numérico) en las tablas en las que participe.
- Para simplificar, tratamos las Fechas como caracteres, por lo que **no** se puede operar con ellas.

**Tabla Empresa:** La cotización tiene 99 como valor por defecto. El Capital debe ser mayor de cero.

**Tabla Tarjeta:** El NumT no puede ser cero. El TipoT debe ser uno de estos : 'VXK','COSA','PISA'

Partimos de este Modelo Relacional de

BDejemplo, simplificado para poder hacer la implementación más sencilla con SQL:

```
Empresa: E(NombreE, Capital, Cotización)
Cliente:  CL(DNI, NombreC, Dirección)
Cliente_N: CL_N(DNI, Provincia, LíneaCredito)
           → (DNI) es FK de Cliente(DNI)
Cliente_I: CL_I(DNI, País, Email)
           → (DNI) es FK de Cliente(DNI)
Moroso:   MO(DNI, NombreC, Dirección)
           → no tiene FK: no todos están en Cliente
Invierte: I(DNI, NombreE, Cantidad, Tipo, CodOferta)
           → (DNI) es FK de Cliente(DNI)
           → (CodOferta) es FK de Oferta(CodOferta)
           → (NombreE) es FK de Empresa (NombreE)
Oferta:   O(CodOferta, Clase, Total)
Crear-Of  no se crea
Tarjeta:  T(NumT, TipoT, Organización)
Compras:  CO(DNI, NumT, NumF, Fecha, Tienda, Importe)
           → (DNI) es FK de Cliente(DNI)
           → (NumT) es FK de Tarjeta(NumT)
Puesto:   P(DNI, Título, Sueldo)
           → (DNI) es FK de Cliente(DNI). Y no necesita otro atr. de CP, es (1,1)
Trabaja:  no se crea
TieneT:   TTA(DNI, NumT, Caducidad, Saldo)
           → (DNI) es FK de Cliente(DNI)
           → (NumT) es FK de Tarjeta(NumT)
TieneTel: TTE(DNI, NumTel)
           → (DNI) es FK de Cliente(DNI)
```

**Solución:** Copiar el contenido del fichero crear-BDejemplo.sql en la Hoja De Trabajo del SQLDeveloper y ejecútalo.

### APARTADO 2: Realizar las siguientes Consultas generales:

```
-- Limpiar "Salida Archivo de Comandos" con el botón de goma de borrar

-- ver el nombre de las tablas que tengo declaradas,
select table_name from tabs;          -- "tabs" es de oracle
-- otra forma con mas información
select * from cat;
-- ver los tipos de los atributos de una tabla (puede que haya más
-- a la dcha, necesita desplazar la barra horizontal)
desc cliente;
-- cuantas filas tengo en Cliente
select count(*) from Cliente;
-- uso de una tabla solo de pruebas llamada 'dual'
select power(4,3) from dual;
select sysdate from dual;
-- Salvar en Archivo todo lo hecho en la sesión:
-- En la solapa "Salida Archivo de Comandos" botón de Guardar
Nombre Archivo z:\.....\prac13.txt
```

### APARTADO 3: Ejecución de consultas. Explicación y Corrección de errores.

Ejecutar una a una las sentencias SQL explicando conceptualmente qué es el resultado, escribiendo en una frase que empieza con '-' en la Hoja de Trabajo:

(ej.: para el (a), "-- todos los clientes, normales y morosos")

Escribir también las respuestas a las preguntas planteadas en cada apartado.

- Limpiar "Salida Archivo de Comandos" con el botón de goma de borrar  
*Como se hizo anteriormente*

a) (select \* from cliente) union (select \* from moroso);

*sale bien*

(select \* from cliente) union (select \* from cliente\_I);

*Da error. ¿por qué?*

b) (select \* from cliente) intersect (select \* from moroso);

*¿ Porqué no sale el cliente 6?*

c) (select \* from cliente) minus (select \* from moroso);

*Qué pasa? porque?*

d) select \* from compras where importe <= all (select importe from compras);

e) select DNI,nombreC

from Cliente

where 'VXK' in (select TipoT

from Compras, Tarjeta

where (Cliente.DNI = Compras.DNI AND

Compras.NumT = Tarjeta.NumT)

);

f) insert into moroso

select \*

from cliente

where dni = '00000001';

g) delete from cliente where dni = '00000005';

*porqué falla? como se arregla?*

h) delete from compras where dni= '00000005';

delete from tienet where dni= '00000005';

i) delete from cliente where dni = '00000005'

*todavía no funciona*

j) 1.- alter table empresa modify (nombreE CHAR(30));

2.- alter table empresa modify (nombreE CHAR(3));

*no se pueden hacer ,decir en cada caso porqué?*

k) alter table empresa modify (capital float);

l) alter table empresa add ( NIF CHAR(8) not null);

*porque no se puede? cual es la solucion si no borramos las tuplas?*

m) alter table empresa modify

( NIF CHAR(8) not null references cliente(DNI));

*¿Porque falla?*

n) create unique index cotind on empresa(cotizacion);

ñ) create unique index dirind on cliente(direccion);

*¿porque falla?*

o) alter table Cliente\_N add CONSTRAINT miAtributo AtributoNuevo Number(6,2);

desc Cliente\_N

p) alter table Cliente\_N drop COLUMN AtributoNuevo;

q) alter table Cliente\_I modify Email VARCHAR2(25);

desc Cliente\_I

r) alter table Invierte drop CONSTRAINT CodOferta not null;

*¿Porque falla?*

o) - Salvar a fichero la sesión: z:\.....

*Como se hizo anteriormente*

#### APARTADO 4: Resolver las consultas siguientes usando SQL.

Para los siguiente ejemplos hay que restaurar los valores originales de la BDEjemplo, ejecutando en otra solapa de la Hoja de Trabajo los dos ficheros:

- 1- El contenido de borrar-BDdejemplo.sql  
para poder repetir esto más fácilmente, salva el contenido en forma de texto desde el editor del SD con el nombre: borrar-BDejemplo.sql  
y así lo puedes abrir desde el menú Editar del SD
- 2- El contenido de crear-BDejemplo.sql

Escribe las sentencias SQL necesarias para obtener lo pedido y ejecutarlas.

Puedes añadir comentarios empezando la línea con "-- explicación : "

a) Limpiar "Salida Archivo de Comandos" con el botón de goma de borrar

*Como se hizo anteriormente*

b) Seleccionar las compras con el mayor importe de todas.

c)- Que inversiones (tuplas) tiene el cliente 2.

d)- Qué clientes (todos los atrib.) están en paro?

e)- Qué clientes (todos los atrib.) no hacen inversiones:

Funciones de Agregación

f)- Cuantas inversiones (tuplas) tiene el cliente 2.

g)- Valor medio de las compras del cliente 5 (use 'avg')

h)- Valor medio de las compras de cada cliente. (use 'avg' y 'group by')

i)- Clientes que gasten en compras como media por encima de 1000 euros

(sobre la consulta anterior aplica 'having')

j)- Que clientes (todos los atrib.) no tienen compras?

k)- Lista de todos los clientes, incluidos los morosos

fin) Guardar la sesión en z:\. . .

## Ejercicios de Modelos Simples para implementar en ORACLE con SQL

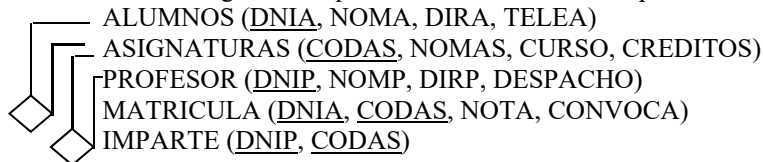
**T5-EJ10** - Disponemos de los siguientes datos de una base de datos que almacena la información de una compañía aérea.

- Personal en general. Un tipo son los pilotos contratados y otros son los administrativos.
- Pilotos contratados, figuran sus datos personales y el tipo de avión que pueden pilotar.
- Los administrativos tienen sus datos personales y el número de despacho.
- La compañía ofrece una serie de trayectos: origen, destino, duración
- De cada trayecto hay programadas una serie de salidas durante el año. Cada salida tiene: cod., fecha, hora
- Pasajeros tienen sus datos personales.
- Un pasajero hace reservas en una fecha para una determinada salida especificando el número de asientos.
- Cada piloto tiene asignado un trayecto fijo durante todo el año.
- Cada trayecto tiene un único piloto.

Se pide:

- Diseñar el modelo entidad-relación (E/R)
- Transformar a un esquema relacional el diagrama entidad-relación obtenido.
- Implementar el esquema relacional en Oracle usando SQL
- Describir en texto posibles consultas que se pueden obtener con esta BD.
- Implementar en SQL las consultas del apartado anterior.
- Escribir ejemplos de los resultados de tuplas obtenidos con las consultas del apartado c)

**T5-EJ11** - Considérese el siguiente esquema relacional de una BD para una universidad:



1.- Implementar dicho esquema en ORACLE con SQL

2.- Escribir en SQL las siguientes consultas:

- Código de las asignaturas en las que está matriculado el alumno cuyo DNI es 100.
- Expediente académico del alumno cuyo DNI es 100. El expediente incluye el nombre, curso y número de créditos de las asignaturas en las que está matriculado el alumno, así como la calificación obtenida.
- Listado de asignaturas con código, nombre y curso que tienen 4,5 créditos .
- Nombre de todas las asignaturas que imparten los profesores del despacho 15.
- Profesores (nombre y apellidos) y las asignaturas que imparten.
- Nota media del alumno 'Pepin Listin'.
- Cuántos alumnos hay matriculados en la asignatura con código 'BD'
- Queremos el DNI de los alumnos con el total de créditos de los que están matriculados

Prácticas Complejas para el Laboratorio en ORACLE usando SQL

- En HOJA APARTE : en directorio Prácticas