

AMPLIACION de BASES DE DATOS

(Profesor : Héctor Gómez Gauchía)

Práctica 1 - SEMANA 1: Corregir y Mejorar el diseño de BDs con Dependencias Funcionales (DFs), Dependencias Multivaloradas (DMs) y Formas Normales (FNs)

Resultado:

- respuestas de cada apartado (todos en un solo fichero word),
- los ficheros .sql de lo que desarrolles en sql y plsql,
- Haz lista de dudas concretas sin resolver sobre tus respuestas para consultar con el profesor

Modo de entrega: No se Entrega

- Si terminas alguno de los apartados indicado en pizarra, avisa al profesor para puntuar la parte de participación en clase
- Los conceptos de esta práctica se evalúan en un examen de control (tal y como se indica en la Ficha Docente), cuya fecha se avisará a tiempo.

NOTA: En casa, puedes instalarte una Máquina Virtual con Oracle: Oracle Express, ver instrucciones en Instalar-MaqVirtual-en-Linux.pdf y otras instrucciones para instrucciones-instal-OracleDB11gR2Express-WIN.pdf

APARTADO 0.- Preparar tu cuenta de Oracle:

- Hacer los pasos indicados en instrucciones-para-ABD.pdf

APARTADO 1.- Reconoce DFs, FNs y consigue un BD en 3FN

Dadas estas dos relaciones que forman una BD:

```
reserva(dni_cliente_reserva, cod_vuelo_cliente_reserva, nomb_cliente_reserva,
apell_cliente_reserva, telefono_cliente_reserva, cuenta_cliente_reserva,
fecha_cliente_reserva, clase_cliente_reserva, nume_plaza_cliente_reserva,
confirmacion_cliente_reserva, cod_ciudad_cliente_reserva,
nombre_ciudad_cliente_reserva)
```

```
vuelo( cod_vuelo, cod_avion, plazas_vuelo_avion, origen_cod_aeropuerto_vuelo,
destino_cod_aeropuerto_vuelo, cod_pais_aeropuerto_origen,
cod_ciudad_aeropuerto_origen, nombre_ciudad_aeropuerto_origen,
direcciónPostal_aeropuerto_vuelo_origen, modelo_avion)
```

Puede que necesites poner otros atributos en alguna tabla teniendo en cuenta que:

El nombre de una ciudad puede estar repetido en dos países.

En el vuelo tenemos, entre otros atributos: el código del avión y la aerolínea.

Se pide hacer lo siguiente. Este apartado tiene varias fases:

Fase 1: En parejas de alumnos (escribe un documento con resultado) → se da un tiempo determinado (se avisa)

- Encuentra la lista de DFs que hay en las dos relaciones.
- Define las tablas necesarias para que toda la BD esté en 3FN (incluyendo las PKs)
- Define dos tablas válidas con más de dos atributos que estén en FN1, pero no estén en la siguiente FN2.
- Haz lo mismo : dos tablas que estén en FN2 pero no en FN3

Fase 2: Compara y discute los resultados con otra pareja → se da un tiempo determinado (se avisa)

- Encuentra DFs distintas entre las dos parejas
- Llegar a un acuerdo y escribe porqué es correcta o porqué falla.

Fase 3: Negocia cuál de las dos parejas ha acertado más y comunica al profesor vuestras conclusiones

APARTADO 2.- (conceptos básicos) Reconocer la clave primaria (CP), clave ajena (CA), claves candidatas (CC), dependencia funcional (DF) y dependencia multivalorada (DM), datos erróneos

Utilizando la *BDejemplo.sql* tal y como está definida, sin CP ni CA, (decide tu cuáles deben ser las CP en los apartados que lo necesites) realiza los siguientes apartados:

2.0- Crea la BDejemplo en Oracle, ejecutando el script *BDejemplo.sql*

Revisa los comentarios dentro del script para entender el significado de los atributos.

2.1- Describe lo indicado en cada apartado:

2.1.x Qué resultado da la siguiente consulta. Indica si es correcto? Qué norma no cumple? Cómo sería correcta?

Consulta: quienes son y donde viven los Clientes activos (tabla *Cliente*) que además son morosos

```
Select Cliente.DNI, Cliente.Direccion
from Cliente, Moroso
where Cliente.Direccion = Moroso.Direccion;
```

2.1.a En la tabla *Cliente*, cuándo podemos considerar CC el atributo Dirección? Describe la respuesta usando la relación que hay entre los valores de los clientes y los valores de las direcciones donde viven.

2.1.b Si, en la tabla *Puesto*, ahora suponemos que cada título tiene el mismo sueldo para todos los clientes que tengan ese título. Indica la DF que provoca redundancia. Cómo se si es problemática?. Cómo la corriges aplicando la teoría? En qué FN queda? Indica los axiomas que cumple para estar en esa FN.

2.1.c Si, en la tabla *Puesto*, ahora suponemos que un cliente tiene más de un Título, cual debe ser la CP? Corrige lo necesario en la BD dentro de Oracle y añade filas con datos inventados para que cumpla lo dicho.

2.2- Qué relaciones se tienen que dar (en cuanto al significado de los atributos) en cada apartado, para que exista la DF indicada:

2.2.X EJEMPLO: Si se cumple en *Empresa*: Cotización → Capital,

Solución: cuando las cotizaciones de las empresas fijan el capital que tienen. Es decir, que para dos Empresas con la misma Cotización tendrán el mismo Capital.

2.2.a - Tabla *Cliente*: NombreC → DNI . Además de decir las relaciones: comprueba en las filas creadas si se cumplen esas relaciones y Corrige las que no las cumplan, inventando datos.

2.2.c - Tabla *Compras*: Tienda → NumF , donde NumF es el número de factura. Además de decir qué significa la DF, indica si es razonable esta DF y porqué?

2.2.d - Tabla *Compras*: NumF → Tienda Además de decir las relaciones, indica si es razonable esta DF y porqué.

2.3- Descubrir la Dependencia y haz un procedimiento para mantener la consistencia de datos:

(se asume que los datos actuales de la BD son correctos)

2.3.a Qué dependencia y de qué clase hay en la tabla *Invierte*, en la siguiente situación:

- Cada Cliente invierte en varias Empresas, pero está obligado a invertir en los mismos Tipos en cada empresa que invierta; y todas las inversiones del mismo Tipo siempre deben tener la misma cantidad.

Esta situación provoca que:

- cada vez que un Cliente invierte en una empresa nueva, tiene que invertir en todos los tipos en los que había invertido antes.
- También sucede que cada vez que el Cliente quiera invertir en otro Tipo diferente a los que ya tenía, le obligan a invertir en todas las empresas (en las que ya tiene inversiones) con ese Tipo y la misma cantidad que tiene en otras inversiones ese mismo Tipo.

----- Práctica 1 - SEMANA 2 -----

2.3.b (mantengo desnormalizada) Implementa un procedimiento *insertoConsistente*, en PLSQL, que vigile la consistencia de datos y que mantenga la dependencia automáticamente cada vez que se inserte una nueva inversión (será el parámetro del proc), manipulando las filas necesarias.

2.3.c (normalizo la tabla) Cómo podrías arreglar la dependencia del apartado anterior dividiendo en tablas? Describe las tablas, no implementes nada.

2.3.d Nos dicen que el número de tarjeta NumT está compuesto por pais, banco, numLocal, así que compras quedaría: Compras: CO(DNI, NumF, Pais, Banco, NumLocal, Fecha, Tienda, Importe)

1.- Describe ahora qué DFs hay ahora en esa tabla, justificando cada una con el significado que tiene.

2.- Obtener el mínimo conjunto de DFs, aplicando los apartados B y C del algoritmo en el documento **tengo-demasiadas-DFs-y-otros-problemas-cierre-minimal.pdf**, tal y como hicimos en clase con el EJ-FN1-f.

3.- Obtener una clave primaria (CP o PK) aplicando el apartado E del mismo documento.

APARTADO 3.- Formas Normales (FN): Reconocerlas, y pasar de una FN a la siguiente

→ Usando de nuevo la BDEjemplo original resuelve los siguientes apartados. Tendrás que determinar las clave primaria antes de hacer el apartado:

3.1 Queremos hacer que la tabla *Cliente* NO esté en la 1ª FN: (Antes indica cual es la CP) Qué relaciones deben cumplir los atributos para que suceda eso. Da ejemplos de filas.

3.2 Introduce una DF en la tabla *Invierte* para que NO esté en la 2ª FN: (Antes indica cual es la CP) Qué relaciones deben cumplir los atributos para que suceda eso (define la CP que usas). Da ejemplos de filas. Cual sería el arreglo?

3.3 Introduce una DF en la tabla *Empresa* para que NO esté en la 3ª FN: (Antes indica cual es la CP) Qué relaciones deben cumplir los atributos para que suceda eso (define la CP que usas). Da ejemplos de filas. Cual sería el arreglo?

3.4 Introduce una DF en la tabla *Tarjeta* para que NO esté en la FNBC: (Antes indica cual es la CP) Qué relaciones deben cumplir los atributos para que suceda eso (define la CP que usas). Da ejemplos de filas. Cual sería el arreglo?