

AMPLIACION de BASES DE DATOS

(Profesor : Héctor Gómez Gauchía)

Práctica 1 – Apdos-1-2-3: Corregir y Mejorar el diseño de BDs con Dependencias Funcionales (DFs), Dependencias Multivaloradas (DMs) y Formas Normales (FNs)

- **Modo de entrega:** No se Entrega

(Se rellena el CUESTIONARIO de la práctica: se avisará cuando esté disponible en el CampusV)

- 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.

- **RESPUESTAS:** En un fichero nuevo word, copia el enunciado de cada apartado en él. Después incluye cada respuesta en el apartado correspondiente. **También:**

- Incluye las filas que están implicadas en cada apartado, ya sea porque estaban provocando un problema o porque las hayas insertado porque te lo pedía.
- Incluye el contenido de los ficheros .sql de los apartados con PLSQL o con instrucciones sql.
- Para organizar bien el fichero, puedes insertar saltos de página.
- Siempre que trabajes, haz Lista de Dudas concretas para consultar con el profesor, online o por email o en clase o en el laboratorio.

→ **Conservar Respuestas para ponerlas en el Cuestionario**

NOTA: los datos de conexión están al final del enunciado, en la misma sección que

Para CONECTARSE al servidor de Oracle de la FDI desde casa

Después sigue debajo, en apartado 0

----- **Práctica 1 - SEMANA 1** -----

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

- Conserva el nombre que tienen los atributos.

Dadas estas dos relaciones que forman una BD: → donde *re* es reserva, *cl* es cliente y *ae* es aeropuerto ←

```
re(dni_cl_re, cod_vuelo_cl_re, nomb_cl_re, apell_cl_re, telefono_cl_re, cuenta_cl_re,
  fecha_cl_re, clase_cl_re, nume_plaza_cl_re, confirmacion_cl_re, cod_ciudad_cl_re,
  nombre_ciudad_cl_re)
```

```
vuelo( cod_vuelo, cod_avion, plazas_vuelo_avion, nomb_ae_origen, dire_ae_origen,
  origen_cod_ae_vuelo, destino_cod_ae_vuelo, cod_pais_ae_origen,
  cod_ciudad_ae_origen, nombre_ciudad_ae_origen, direcciónPostal_ae_vuelo_origen,
  modelo_avion, cod_aerolínea)
```

→ Puede que necesites repetir algunos atributos en alguna tabla.

→ Ten en cuenta estos requisitos:

El mismo nombre y código de una ciudad puede existir en dos países.

En el vuelo debes tener la aerolínea. El código de vuel es único, e independiente de la línea.

→ Si necesitas asumir otros requisitos (atributos nuevos, DFs nuevas) cuando veas más de una interpretación: escoge la mejor y describe textualmente lo que asumes (los supuestos) en este punto del documento (aquí) a continuación (como los requisitos anteriores).

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

1.- Encuentra la lista de DFs que hay en las dos relaciones,

→ **ver al final de este doc: APTDO 1: AYUDAS par Encontrar DFs y obtener tablas en 3ªFN**

2.- Define las tablas necesarias para que toda la BD esté en 3FN (incluyendo las PKs y las FKs)

APARTADO 2.- Reconocer la clave primaria (CP), clave ajena (CA), claves candidatas (CC), dependencia funcional (DF) y dependencia multivalorada (DM), datos erróneos. Escribir procedimientos sencillos en PLSQL.

→ Utilizando la *BDejemploSimple.sql* tal y como está definida, sin CP ni CA, realiza los siguientes apartados: (en algunos apartados, debes decidir tu cuáles son las CP para escoger la respuesta)

2.0- Crea la *BDejemplo* en Oracle, ejecutando el script *BDejemploSimple.sql* tal y como está. Revisa los comentarios dentro del script para entender el significado de los atributos.

2.1- Describe lo indicado en cada apartado: (SIEMPRE contesta cada pregunta por separado)

2.1.x - 1.- Qué resultado da la siguiente consulta?.

2.- Indica si ese resultado es correcto?

3.- Qué norma no cumple la unión/ join?

4.- Cómo sería correcta? Para comprobar, añade el dni de moroso al principio de la select.

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. 1.- Indica la DF que provoca redundancia. 2.- Inserta filas que provoquen la redundancia (usa "insert"). 3.- Cómo se si es problemática?. 4.- Cómo la corriges aplicando la teoría?. 5.- En qué FN queda la tabla?. 6.- 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. 1.- Cual debe ser la CP? 2.- Corrige lo necesario en la BD dentro de Oracle con "Alter" 3.- Inserta filas con datos inventados para que cumpla lo dicho. (usa "insert").

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 resuelto: Si se cumple en *Empresa*: Cotización → Capital,

Solución: cuando las cotizaciones de las empresas determinan 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 . 1.- Decir las relaciones. 2.- Comprueba en las filas creadas si se cumplen esas relaciones. 3.- Corrige con *update* las que no las cumplan, cambiando sus valores (inventa otros).

2.2.c - Tabla *Compras*: Tienda → NumF , donde NumF es el número de factura. 1.- Decir qué significa la DF. 2.- indica si es razonable esta DF y porqué?

2.2.d - Tabla *Compras*: NumF → Tienda 1.- Decir las relaciones. 2.- Indica si es razonable esta DF y porqué?.

2.3- Descubrir la Dependencia y haz un procedimiento para mantener la consistencia de datos. Pasos: (se asume que los datos actuales de la BD son correctos)

2.3.a 1.- Qué dependencia y de qué clase hay en la tabla *Invierte*, dadas estas reglas de un Broker para invertir: - 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.

→ 2.- Inserta filas con datos inventados para que se cumpla esa dependencia.

----- Práctica 1 - SEMANA 2 -----

IMPORTANTE: para estos apartados necesitamos tener la BD como estaba al empezar, para ello:

BORRAR la BD (script borradb.sql) y crearla de nuevo (script BDejemploSimple.sql) Comprueba que no ha dado ningún error.

2.3.b (mantengo desnormalizada) Implementa un procedimiento *insertoConsistente*, en PLSQL, que vigile la consistencia de datos y que mantenga la dependencia del apartado 2.3.a automáticamente cada vez que se inserte una nueva inversión (sus datos son parámetros del proc), actualizando o creando las filas necesarias.

Utiliza, como base para implementar el archivo **/para-Apdo-2.3.b/INSERTOCONSISTENTE-Esqueleto.sql** . Para probar el procedimiento, ejecuta **/para-Apdo-2.3.b/prueba-insertconsistente.sql**.

2.3.c (normalizo la tabla) Cómo podrías arreglar la dependencia del apartado anterior dividiendo en tablas? Describe las tablas, sin implementar nada.

2.3.d (para NOTA) En otra situación, nos dicen que el número de tarjeta NumT está compuesto por país, banco, número de Cuenta Local de ese banco. Quedando la tabla así Compras:

CO(DNI, NumF, País, Banco, CuentaLocal, Fecha, Tienda, Importe)

- 1.- Describe textualmente qué DFs hay ahora en esa tabla, justificando cada una con el significado que tiene.
- 2.- Obtener el mínimo conjunto de DFs solo de esa tabla, 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. (solo usando el conjunto de DFs en Compras).
- 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: 1.- indica cual es la CP. 2.- Qué relaciones deben cumplir los atributos para que suceda eso. 3.- Da ejemplos de filas que cumplan lo dicho.
- 3.2** Introduce una DF en la tabla *Invierte* para que NO esté en la 2ª FN: 1.- Indica cual es la CP. 2.- Qué relaciones deben cumplir los atributos para que suceda eso (define la CP que usas). 3.- Da ejemplos de filas que cumplan lo dicho. 4.- Cual sería el arreglo?
- 3.3** Introduce una DF en la tabla *Empresa* para que NO esté en la 3ª FN: 1.- Indica cual es la CP. 2.- Qué relaciones deben cumplir los atributos para que suceda eso (define la CP que usas). 3.- Da ejemplos de filas que cumplan lo dicho. 5.- Cual sería el arreglo?
- 3.4** Introduce una DF en la tabla *Tarjeta* para que NO esté en la FNBC: 1.- Indica cual es la CP. 2.- Qué relaciones deben cumplir los atributos para que suceda eso (define la CP que usas). 3.- Da ejemplos de filas que cumplan lo dicho. 4.- Cual sería el arreglo?

APTDO 1: AYUDAS par Encontrar DFs y obtener tablas en 3ªFN

.org

- 1.- Escribe en una línea aparte cada atributo que diferencie una entidad (serán códigos únicos)
- 2.- Recorrer el resto de atributos y colocar los que dependan de cada código escrito en el paso anterior.
 - A veces el mismo código estará en dos DFs , cuando sea PK y FK
- 3.- Si hay dudas, escribir un requisito indicando lo que asumes
- 4.- Cuando estén todos los atributos colocados (dependiendo de códigos)
 - Dentro de cada grupo formado por un código y sus dependientes
Busca nuevas dependencias entre ellos.
- 5.- .- Escribe para cada código una tabla y su PK. Debe tener sentido en conjunto
- 6.- Comprueba que cada tabla está en 3ª FN
 - Asegúrate que tienes el cierre minimal de las DFs de esa tabla
 - Comprueba los dos axiomas axiomas de 3ª FN.
 - Si no está en 3ªFN, descomponla en dos tablas y
- comprueba que las nuevas están en 3ªFN.
- 7.- Comprueba que tienen sentido las tablas obtenidas:
 - Ejemplo:
 - Los atributos del mismo concepto (código) están en una tabla.
 - Quiero incluir algún atributo más por decisión propia.
 - Si tienes muchas tablas de dos atributos quizá conviene desnormalizar
 - Para agrupar tablas del mismo "concepto".
 - Solo si quieres sacrificar la 3ªFN y la vas a controlar programando

- 1.- Instalar el programa GlobalProtect para conectarte a la red virtual de la UCM (la VPN Galería)
Mirar en los servicios informáticos de la UCM:
<https://ssii.ucm.es/faq/conexion-vpn/como-configurar-una-conexion-vpn-a-la-red-ucm>
 - 2.- instalar el SQLdeveloper desde Web de Oracle
(el que incluye el JDK 11, te evita problemas si tienes activa otro JDK diferente)
(Importante: **NO instales** ningún servidor Oracle , Oracle XE ni máquina virtual)
 - 3.- **Conectarse al GlobalProtect** Siempre: ANTES de abrir el SQLdeveloper
Si no lo haces: se queda colgado
 - 4.- Dentro de tu SQLdeveloper , crea una conexión con
botón DCHO en icono "Oracle conexiones" con estos datos:
 - Nombre de conexión : tunnel-Oracle-FDI (cualquier nombre que te recuerde conectar el túnel)
 - TU USUARIO del LABORATORIO
 - si NO tienes usuario personal que debes haber creado en la Facultad :
 - crea una conexión con ADMINUSER,(ver NOTA para la clave) solo para :
 - después crear TU USUARIO : haz TODAS las *instrucciones-para-ABD.pdf*
 - Si ya tienes usuario: crea una conexión con tu usuario para entrar fácilmente:
 - TU CONTRASEÑA
 - Rol: por defecto

Tipo de Conexión: Básico

 - Host: 147.96.85.48
 - Puerto 80
 - SID: bd1

NOTA para la clave: si se te ha olvidado la contraseña de tu usuario:

 - vete a la /prac1-apdos-1-2-3-Normalizar-CV.zip :
 - en un archivo aparte: adminuser.txt, está la clave para el ADMINUSER, desde el cual puedes cambiar la clave de tu usuario según las instrucciones *Instrucciones-para-ABD.pdf* en la misma práctica.
 - 5.- A continuación sigue los pasos de la PRAC1, APARTADO 0 : *Instrucciones-para-ABD*
-

NO NECESITAS INSTALAR EL SERVIDOR, → → solo necesitas el SQLdeveloper para la conexión desde casa

hay tres enfoques para instalar el Servidor en tu PC de casa (WIN/MACOS):

IMPORTANTE: todas las prácticas se deben ejecutar en el servidor de FDI, aunque instales el servidor en tu PC, POR ESO RECOMIENDO NO INSTALAR EL SERVIDOR en TU PC.

1. Lo más parecido a la instalación de la Facultad es instalar Oracle Express(servidor) y el SQLdeveloper (cliente)
2. Si no quieres tener directamente Oracle en tu S.O., antes puedes instalar una Máquina Virtual y luego hacer los pasos anteriores:
<https://www.oracle.com/virtualization/technologies/vm/downloads/virtualbox-downloads.html>
<https://www.virtualbox.org/manual/UserManual.html>
3. Todavía hay otra opción: bajarte una MV que ofrece Oracle con la BD y el SQLdeveloper
<https://www.oracle.com/database/technologies/databaseappdev-vm.html>
(te pide login, creas una cuenta y sigues)