# 2º Grado en Ingeniería Informática

Asignatura: Bases de Datos Curso: 2015/16

Facultad de Informática

# PRÁCTICA P2. Definición y modificación de datos en SQL

# **Objetivos**

• Construir, depurar y ejecutar sentencias SQL de definición de datos (LDD): creación, eliminación y alteración de la estructura de los elementos que componen un esquema de base de datos.

• Construir, depurar y ejecutar sentencias SQL de modificación de información almacenada en una base de datos: inserción, actualización y borrado (LMD).

#### **Contenidos**

Esta práctica consiste en la elaboración de sentencias SQL para **crear** y **destruir** elementos de un esquema de base de datos determinado y para **alterar** su estructura (LDD, lenguaje de definición de datos), así como para **introducir**, **eliminar** y **actualizar** datos almacenados en las tablas del mismo (LMD, lenguaje de manipulación de datos).

# Normas de realización

Es necesario contestar los ejercicios en el <b>orden</b> indicado en el enunciado.
Es posible que algún apartado necesite <b>más de una sentencia</b> SQL para ser resuelto; en tal caso hay que escribirlas en el orden en que deben ser ejecutadas para que funcionen correctamente.
<b>Si</b> lo que se pide en un ejercicio <b>se puede hacer</b> directamente <b>en Oracle</b> (la <u>versión 11g</u> es la utilizada para realizar las prácticas), hay que hacerlo y con ello ya está resuelto el ejercicio. Sin embargo, <b>si</b> algún ejercicio <b>no se puede hacer directamente en Oracle</b> hay que
<ul> <li>Justificar por qué no se puede hacer en Oracle.</li> <li>Indicar cómo se haría en el estándar SQL-92.</li> </ul>
<ul> <li>Simularlo en Oracle (por ejemplo, con varios INSERT, UPDATE, DELETE ejecutados en cierto orden, con SELECT si se trata de simular una restricción de integridad, etc.).</li> </ul>
Se debe proponer <b>distintas soluciones</b> de un ejercicio cuando haya lugar a ello. Es habitual que exista una solución que sólo use sentencias LMD y otra que también use sentencias LDD <mark>1</mark> .
Es imprescindible que <b>cualquier restricción de integridad</b> (referencial, de dominio, de tabla, etc.) existente antes de un determinado ejercicio, <b>siga existiendo después</b> de ejecutar el conjunto de sentencias que dan solución a dicho ejercicio.
Debe <b>justificarse</b> adecuadamente toda <b>decisión</b> tomada a la hora de realizar cada apartado.
Además, debe indicarse el efecto que cada sentencia tiene sobre el esquema de base de datos.
Durante la <b>realización de la práctica en los laboratorios</b> , la estructura y/o contenido de los <b>elementos</b> de base de datos (esquema, tablas, vistas, índices, etc.) ha de ser <b>creado</b> , <b>modificado o alterado</b> en la cuenta Oracle de su grupo de prácticas ( <b>bdgxxxx</b> ).
En el <u>momento de la entrega del informe</u> , las tablas de base de datos correspondientes a su grupo de prácticas deben estar <b>creados y</b> contener <b>datos</b> (es decir, haber sido ejecutados sólo los ejercicios

### Fecha de entrega

La práctica se puede entregar hasta el viernes 15 de abril a las 23:59h. Es decir, en el Aula Virtual la Tarea correspondiente, denominada de forma similar a "Entrega de Práctica P2. Definición y Modificación de datos en SQL", se cerrará en ese momento.

1 y 2), de modo que el esquema estará listo para la ejecución correcta del 3º al último ejercicio.

<sup>1</sup>¿Ya tienes claro qué significa LDD y LMD y las diferencias entre las sentencias de cada tipo? ;-)

# **Enunciado: "ONG"**

La coordinadora nacional de organizaciones no gubernamentales (ONG) desea mantener una base de datos de las asociaciones de este tipo que existen en nuestro país. Para ello necesita almacenar la información sobre cada ONG, los socios que las componen, los proyectos que realizan y los trabajadores de las mismas. Las cuestiones más importantes que hay que tener en cuenta son:

- De cada ONG se desea almacenar su código, denominación, provincia, teléfono y dirección de correo electrónico, además de su campo de acción ('derechos humanos', 'nutrición y seguridad alimentaria', 'medioambiente', 'inmigración', 'cooperación para el desarrollo', 'discapacidad', 'drogodependencia', etc.).
- Cada ONG está formada por socios, de los que se precisa conocer su DNI, nombre, la fecha de alta en la ONG y la cuota mensual con la que colaboran. Es posible que una persona esté asociada al mismo tiempo a varias ONG diferentes.
- Los trabajadores de estas organizaciones pueden ser contratados o voluntarios. De cada trabajador se precisa recoger su DNI, su nombre, la fecha en que ingresaron en la organización, su año de nacimiento, su profesión y las horas que dedica a la ONG.
- Los voluntarios trabajan en la organización de forma desinteresada.
- Los contratados son trabajadores que perciben un sueldo. Cada ONG tiene un responsable que se encuentra entre sus trabajadores contratados. Un trabajador de cierta ONG no puede trabajar en otra, ni siquiera de voluntario.
- Un socio de una ONG concreta no puede ser trabajador de la misma.
- Las ONG llevan a cabo diversos proyectos. De cada proyecto interesa almacenar su número de identificación dentro de cada ONG, el objetivo que persigue, el número de beneficiarios a los que afecta y el país y zona concreta dentro de éste en el que se lleva a cabo.
- Un proyecto consta de varias acciones o actividades (por ejemplo, el proyecto denominado 'Mejora de las alternativas económicas y sociales para la población campesina de Cayambe, en Ecuador' incluye las siguientes acciones: '1) Puesta en marcha y funcionamiento de un centro de atención a los trabajadores que ofrezca servicio sanitario, legal y de actividades recreativas; 2) Formación de los trabajadores; 3) Sensibilización general de los trabajadores, autoridades y empresarios; 4) Funcionamiento de un comité interinstitucional para una gestión adecuada de la floricultura; y 5) Desarrollo de alternativas productivas asociativas agropecuarias y microempresas. ').
- Los trabajadores de las ONG pueden ser asignados a proyectos, e interesa conocer en qué acciones concretas participan. Un trabajador puede trabajar en acciones de diferentes proyectos siempre que todos éstos se desarrollen en el mismo país.

# Esquema relacional

Como resultado de las etapas de diseño conceptual y diseño lógico de la base de datos, se ha identificado el siguiente conjunto de tablas que se muestra a continuación en orden alfabético.

Las <u>claves primarias</u> aparecen subrayadas. Las *claves ajenas* (externas) se muestran en cursiva y las correspondientes restricciones de integridad referencial pueden deducirse mediante el nombre de las columnas relacionadas (clave ajena-clave primaria) y el nombre de la tabla referenciada.

```
ACCION (<u>ong, idproyecto, idaccion</u>, descripcion)

COLABORACION (<u>ong, socio</u>, fechaalta, cuota)

ONG (<u>codigo</u>, nombre, email, telf, provincia, campo, <u>responsable</u>)

PARTICIPACION (<u>ong, idproyecto, idaccion, trabajador</u>)

PROYECTO (<u>ong, idproyecto</u>, objetivo, pais, zona, numbeneficiarios)

SOCIO (<u>dni</u>, nombre)

TRABAJADOR (<u>dni</u>, nombre, ong, fechaingreso, esvoluntario, nacimiento, profesion, horas, sueldo)
```

# **Ejercicios**

#### 1. Creación de las tablas.

- Se aconseja **dibujar el esquema referencial** correspondiente al esquema de base de datos, utilizando una notación gráfica similar a la incluida en el enunciado de la práctica **P1** de consultas. Se recomienda dibujar este esquema ANTES de intentar crear las tablas, para tener una visión clara de las relaciones existentes entre ellas (integridad referencial).
- Antes de crear las tablas, también es conveniente...
  - Elegir el **tipo de datos** y **longitud** más adecuados para las columnas de las tablas.
  - <u>Deducir</u> restricciones de integridad de columna y tabla, y darles un nombre adecuado:
    - Valores por defecto (DEFAULT)
    - Atributos no nulos (NOT NULL)
    - Clave primaria (PRIMARY KEY)
    - Clave alternativa (UNIQUE)
    - Otras restricciones (CHECK)
    - Claves ajenas o externas (FOREIGN KEY... REFERENCES...) **<imbeddings** importante!
      - Para cada clave ajena, elegir y justificar las acciones de mantenimiento de la integridad referencial (ON DELETE <acción> y ON UPDATE <acción>) que se consideren más adecuadas: **Definir todas como NO ACTION** e indicar como comentarios cuáles son en realidad y a su juicio, las más apropiadas.

Consejo: se recomienda dar un nombre a las CONSTRAINT correspondientes a los 4 últimos.

- **Crear las tablas en el orden adecuado**. En este sentido, es imprescindible **resolver el problema** que surge al existir un **ciclo referencial** entre las tablas ONG y TRABAJADOR.

**Importante:** el CREATE TABLE de cada tabla debe incluir la definición de sus restricciones de integridad de columna y de tabla (incluidas, por supuesto, las de integridad referencial o claves externas). Sólo cuando esto no sea posible se añadirán a posteriori dichas restricciones de integridad con ALTER TABLE ADD...;

### 2. Insertar varias filas en todas las tablas del esquema.

- Se valorará la **calidad** del conjunto de filas insertadas en cada tabla (es decir, que se cubran todos los casos posibles). Cada INSERT debe incluir la lista de nombres de columnas antes de la cláusula VALUES.
- Las inserciones hay que hacerlas en el **orden adecuado** de forma que no se violen las reglas de integridad referencial. Se debe solucionar los problemas que puedan surgir, precisamente, por la comprobación de la integridad referencial. En este sentido, hay que indicar y justificar la mejor solución a los siguientes problemas (si ocurren):
  - Cuando para insertar en una tabla se necesita que ya haya datos almacenados en otra tabla.
  - Introducir filas en una tabla que contiene una autorreferencia.
  - Si dos tablas se referencian mutuamente y no se puede insertar en ninguna de las dos sin existir datos en la otra. Es decir, cuando existe un ciclo referencial.

Algunas de las opciones que se puede considerar en los problemas anteriores son las siguientes:

- Diferir la comprobación de la restricción de integridad referencial y después volver a restaurar su inmediatez.
- Eliminar una restricción de integridad referencial de una tabla T, para después volver a crearla.
- Desactivar o inhabilitar una restricción de integridad referencial y después volver a reactivarla.
- Inhabilitar una restricción NOT NULL (no la de integridad referencial) de una clave ajena.

. . .

### 3. Insertar comentarios en una tabla y en una columna.

- Crear en la BD los comentarios (sentencia COMMENT ON) de la tabla ONG, y de todas las columnas de la tabla ONG.
- Comprobar la creación correcta de los comentarios anteriores, mostrando por pantalla un par de SELECT sobre las vistas del diccionario de datos denominadas USER\_TAB\_COMMENTS y USER\_COL\_COMMENTS.
- ¿En Oracle la definición de un comentario se considera una sentencia en lenguaje de definición de datos (LDD)? ¿Qué implicaciones tiene la ejecución de una sentencia de este tipo?

# 4. Modificar una clave primaria de manera ordenada: cambiar el DNI de una persona.

Elegir una persona cualquiera que sea socio de varias ONG y que, a la vez, esté contratado por otra ONG tal que el trabajador participe en diversas acciones dentro de los proyectos de esta última ONG.

- Cambiar su DNI por un nuevo valor inexistente hasta este momento.

Hay que tener en cuenta que el DNI de un trabajador es la clave primaria de la tabla TRABAJADOR, y que puede estar referenciado desde otras tablas, las cuales se ven afectadas por el cambio.

Además, puesto que el trabajador también es socio de otras ONG; es imprescindible reajustar adecuadamente su DNI en SOCIO.

**Importante**: En este ejercicio hay que considerar que todas las claves ajenas implicadas tienen **ON UPDATE NO ACTION** (independientemente de las decisiones tomadas para realizar el ejercicio **1**).

#### 5. Intercambiar los socios de dos ONG.

- Elegir dos ONG cualesquiera con un número de socios distinto.
- Mostrar con un SELECT la clave primaria, nombre y el número de socios de cada ONG.
- Intercambiar los socios de ambas ONG, de forma que los que antes pertenecían a una pasen a ser socios de la otra y viceversa. Se puede utilizar una o varias sentencias UPDATE.
- Volver a mostrar los datos anteriores para comprobar que se ha realizado el intercambio.
- Por último, hacer ROLLBACK para dejar los datos como estaban al principio (antes de este ejercicio) y comprobar si ha sido así (mostrarlos). Comentar este resultado.

#### Importante:

En este ejercicio hay que considerar que todas las claves ajenas implicadas tienen **ON UPDATE NO ACTION** (independientemente de las decisiones tomadas en el ejercicio **1**).

<u>No se puede</u> variar la estructura actual (columnas, por ejemplo) de las tablas, ni se puede usar tablas temporales intermedias, ni utilizar estructuras **CASE** ni **DECODE**.

Para resolver este ejercicio hay que seguir las indicaciones de la 3ª norma de realización mostrada en este enunciado: si algo no se puede implementar directamente en Oracle, a) justificar por qué, b) resolverlo en SQL-92, y c) simularlo en Oracle.

#### 6. Borrar una ONG.

**Importante**: En este ejercicio, hay que ASEGURARSE de que todas las claves ajenas implicadas tienen **ON DELETE NO ACTION** (independientemente de las decisiones tomadas para realizar el ejercicio **1**).

Elegir una ONG con varios trabajadores (entre los que se encontrará el responsable), que posea diversos proyectos, cada uno con varias acciones en las que participen unos cuantos trabajadores. Además, la ONG debe tener entre sus socios algunos que también colaboren con otras ONG.

- Una vez elegida la ONG, borrar todos los datos referentes a la misma.

Puesto que una ONG pueden hacer referencia a filas de otra tabla, y puede estar referenciada desde otras tablas, es importante razonar qué se hace con la información afectada por los borrados. Por ejemplo, hay que considerar a) ¿Qué ocurre con los socios de dicha ONG? ¿Y si esos socios lo son también de otras ONG? b) ¿Qué hacer con los registros referentes a los proyectos y las acciones puestos en marcha por la ONG? c) ¿Qué pasa con los trabajadores de esa ONG?

### 7. Crear y manipular una vista.

- <u>Definir</u> una vista llamada VOLUNTARIO que muestre el nombre, nombre de su ONG, profesión y horas de los trabajadores voluntarios. Debe tener 4 columnas: nombre, ong, profesion y horas.
- Mostrar el contenido completo (con SELECT \*) de la vista, ordenado por ong y voluntario.
- Eliminar de la vista la columna profesion.
- Mostrar de nuevo el contenido de la vista para confirmar si se ha realizado el cambio o no.
- <u>Modificar</u> la vista para que los valores de la columna horas aparezcan incrementados en 2 horas.
- Volver a mostrar el contenido completo de la vista, para comprobar la corrección de los datos.
- ¿Qué datos se han modificado realmente?
- Si se inserta un nuevo voluntario ¿se aplica el cambio al visualizar la vista?

**Importante:** Para resolver este ejercicio hay que seguir las indicaciones de la 3ª norma de realización mostrada en este enunciado: si algo no se puede implementar directamente en Oracle, a) justificar por qué, b) resolverlo en SQL-92, y c) simularlo en Oracle.

### 8. Crear y cargar una tabla, y modificar su estructura.

- Crear una nueva tabla llamada CONTRATADO como una copia de la tabla TRABAJADOR. Debe tener las mismas columnas, con iguales tipos de datos que TRABAJADOR, <u>salvo</u> las columnas esvoluntario, nacimiento, profesion y horas que desaparecen, y se debe <u>añadir una columna</u> para el número de veces que el trabajador participa en acciones de proyectos de la ONG (participaciones).

Además, CONTRATADO sólo debe almacenar los datos relativos a trabajadores que son socios de <u>dos o menos ONG</u>.

**Importante**: La tabla ha de ser <u>simultáneamente creada y cargada con datos</u> extraídos de los existentes en otras tablas del esquema.

- Mostrar con un SELECT el contenido de la tabla CONTRATADO.
- Añadir a la tabla una columna llamada jefe, que contendrá el <u>nombre del responsable</u> de la ONG en la que está contratado cada trabajador. Elegir tipo de datos y tamaño adecuados. El valor por defecto será SIN JEFE.
- Utilice una sentencia para actualizar el valor de la columna jefe (simultáneamente para todas las filas) con el nombre del responsable de cada trabajador con base en su ONG.
- Volver a mostrar (con SELECT) el contenido de la tabla, y comprobar que los datos son correctos.
- Eliminar la columna jefe de manera que CONTRATADO quede como tras su creación.

### 9. Restricciones de integridad. Establecer las siguientes restricciones de integridad:

- RI1. Un trabajador sólo puede participar en proyectos de su ONG.
- RI2. Una persona no puede ser socio y trabajador de la misma ONG.
- RI3. Un trabajador no puede ser responsable de más de una ONG.
- RI4. El responsable de una ONG ha de ser uno de entre los trabajadores contratados por tal ONG.
- RI5. Un socio debe serlo al menos de una ONG.

#### Importante:

- Si alguna restricción ya ha sido implementada antes, quizá en una sentencia de creación de tabla, en este ejercicio hay que indicar claramente cómo se ha implementado.
- Para resolver este ejercicio hay que seguir las indicaciones de la 3ª norma de realización mostrada en este enunciado, especialmente para aquellas restricciones que no es posible implementar directamente en Oracle: a) justificar por qué no se puede, b) resolverlo en SQL-92, y c) simularlo en Oracle.
- La simulación de una restricción que no es posible implementar directamente en Oracle puede consistir, por ejemplo, en una SELECT que muestre las filas que violan la restricción de integridad correspondiente.
- Muy importante: **No** hay que programar disparadores (*triggers*).

# Documentación que se debe entregar

La entrega se realizará mediante el **Aula Virtual**, antes de la fecha límite indicada en la **Tarea** correspondiente. Es importante recordar que debe hacerlo cada uno de los miembros de cada grupo de prácticas. Se debe entregar el **informe** de realización de la práctica (memoria), así como el conjunto de **guiones** (*scripts*) SQL elaborados.

Todos estos ficheros deben ser incluidos en un **fichero comprimido** con el nombre **bdgxxxx-p2** (.zip o .rar), sustituyendo **bdgxxxx** por el nombre de su grupo de prácticas.

El **nombre y formato** de cada uno de los ficheros será el siguiente:

(1) Informe de la práctica. Documento llamado (en minúsculas) bdgxxxx-p2.ext La extensión (.ext), es decir, el formato del documento, puede ser .pdf (recomendado) o bien .doc, .docx, .rtf, .odt.

El informe debe tener las páginas numeradas y debe incluir lo siguiente;

- ☐ Portada, que muestre estos datos:
  - asignatura (Bases de Datos), curso académico (20nn/nn) y convocatoria (junio, julio, febrero).
  - identificador (P2) y nombre de la práctica (Definición y modificación de datos en SQL).
  - nombre del **grupo** de prácticas (**bdgxxxx**), **nombre** y **apellidos** de cada componente.
  - dirección de **correo electrónico** de cada integrante del grupo de prácticas.
- ☐ Guión (script) de creación del esquema de base de datos empleado en la práctica, en SQL de Oracle.
- ☐ Para cada ejercicio deberá aparecer lo siguiente:
  - número, título y enunciado de ejercicio.
  - comentarios y aclaraciones que se considere necesarios.
  - **texto de las** *sentencias* (LDD y/o LMD) necesarias para resolver el ejercicio, en el orden en el que deben ser ejecutadas, con las <u>indentaciones adecuadas</u> (márgenes, sangrías) y fácilmente <u>legible</u>.
  - ejemplos que se considere oportunos para comprobar el correcto funcionamiento de la solución propuesta para el ejercicio.

Es posible presentar varias soluciones alternativas del mismo ejercicio. Se puede indicar la alternativa que se considera más adecuada.

- **(2) Guiones** (*scripts*) **SQL Oracle**. Ha de crearse un guión (fichero de texto plano) por <u>cada</u> ejercicio, llamado (en minúsculas) **bdgxxxx-p2-nn.sq1** (donde **nn** es el nº de ejercicio), que contendrá...
  - número y título del ejercicio.
  - texto de las sentencias LDD y/o LMD, en el orden en el que deben ser ejecutadas, con las indentaciones adecuadas y legible.
  - NO DEBE INCLUIR ejemplos de comprobación ni el resultado de la ejecución de las sentencias.

### Criterios de evaluación

- Es obligatorio entregar el **informe** de la práctica y todos los **guiones** (*scripts*) SQL. Además, es imprescindible presentar la solución de **todos los ejercicios**. No se corregirá la práctica si no se ha presentado la documentación completa.
- La lectura del **informe** ha de ser **suficiente** para la comprensión total del trabajo realizado en la práctica, sin necesidad de recurrir a los guiones.
- No es suficiente con que "las sentencias funcionen", esto es, que se consigan los resultados solicitados "sea como sea", sino que se tendrá en cuenta el estilo en la resolución de cada consulta (uso de **buenas prácticas** en el uso de SQL). Se valorará la explicación de la resolución de cada ejercicio, la presentación de varias alternativas para resolver los ejercicios y la justificación de las mismas, así como la estructura de la documentación explicativa (informe) de la práctica.

Es imprescindible respetar estrictamente las normas y el formato de presentación del informe de la práctica, detallados en este documento