ESTUDIO

**Aprendiendo a crear, actualizar y borrar tablas**

a adicionar, actualizar y eliminar tuplas

a crear y borrar vistas

a crear y borrar índices

consultar los datos sobre los datos

**Revisando los tipos de datos**

**Completen la siguiente tabla de equivalencia de tipos de datos. Escriba los tipos correspondientes en el lenguaje del modelo conceptual, en SQL y en SQL ORACLE.**

|  |  |  |
| --- | --- | --- |
| **Modelo** | **SQL:2008** | **ORACLE** |
| ENTERO(N) | INT | INTEGER(N) |
| REAL (D, M) | REAL | NUMBER(D,M) |
| CARÁCTER | CHAR | CHAR(N) |
| CADENA(N) Fija Flexible | VARCHAR(N) | VARCHAR2(N) |
| HORA FECHA HORA+FECHA | Datetime | DATE |

**3. Aprendiendo a definir las restricciones sobre atributos y de clave de forma independiente**

**A. Investigue la sentencia ALTER TABLE ADD CONSTRAINT que le va a permitir adicionar las**

**restricciones de atributos y claves a las tablas con nombre separando la creación de tablas de la**

**definición de restricciones.**

Con ALTER TABLE ADD CONSTRAINT podemos agregar una restricción de clave externa, clave unica, clave primaria y de verificación

Sintaxis clave primaria:

ALTER TABLE <name>

ADD CONSTRAINT <constraint\_name>

PRIMARY KEY (<column\_name>)

Sintaxis clave unica:

ALTER TABLE <name>

ADD CONSTRAINT <constraint\_name>

UNIQUE (<column\_name>)

Sintaxis clave unica:

ALTER TABLE <name>

ADD CONSTRAINT <constraint\_name>

FOREIGN KEY (<column\_name>) REFERENCES <name2>(<column\_name2>);

**B. ¿Cómo quedarían definidas las siguientes tablas separando la definición de datos de la definición de**

**restricciones? (Un ALTER TABLE por cada restricción)**

|  |  |
| --- | --- |
| **CREATE** **TABLE** games (  yr **INT** **NOT** **NULL** **PRIMARY** KEY,  city **VARCHAR**(20) **UNIQUE** );  La restricción de llave primaria se debe llamar PK\_GAMES  La restricción de llave única se debe llamar UK\_GAMES | **CREATE** **TABLE** games (  yr **INT** **NOT** **NULL**,  city **VARCHAR**(20));  **ALTER** **TABLE** games **ADD** **CONSTRAINT** PK\_GAMES **PRIMARY** **KEY** (yr);  **ALTER** **TABLE** games **ADD** **CONSTRINT** UK\_games  **UNIQUE** **KEY**(city); |
| **CREATE** **TABLE** Orderses (  O\_Id **int** **NOT** **NULL** **PRIMARY** **KEY**,  OrderNo **int** **NOT** **NULL**,  P\_Id **int** **FOREIGN** **KEY** **REFERENCES** Persons(P\_Id) ) ;  La restricción de llave primaria de debe llamar PK\_ORDENES La restricción de llave foránea debe llamarse FK\_ORDENES\_PERSONAS | **CREATE** **TABLE** Orderses (  O\_Id **int** **NOT** **NULL**,  OrderNo **int** **NOT** **NULL**,  P\_Id **int** ) ;  **ALTER** **TABLE** Orderses **ADD** **CONSTRAINT** PK\_ORDENES **PRIMARY** **KEY**(O\_id);  **ALTER** **TABLE** Orderses **ADD** **CONSTRAINT** FK\_ORDENES\_PERSONAS **FOREIGN** **KEY** (P\_id) **REFERENCES** persons (P\_id) |

**B. INVESTIGANDO SQL Developer**

Considerando la herramienta SQL Developer

**A. Investigue las funcionalidades básicas de la herramienta.**

**Crear conexiones:** Podemos crear y testear tantas conexiones a bases de datos como queramos. Destaca la capacidad que tiene para conectarse a otro tipo de bases de datos aparte de Oracle. Se permite almacenar los datos de conexión a todas las BBDD que queramos. Mediante el parámetro “Tipo de Conexión” podemos elegir introducir los parámetros de la conexión de forma manual, seleccionar una de las BBDD definidas en el fichero tnsnames.ora o bien introducir los datos con formato URL de JDBC.

**Modelo E/R**

Se puede realizar mediante Herramientas - Data Modeler.

Es necesario haber creado previamente las claves primarias y ajenas necesarias. Muestra una estructura de las relaciones, pero no permite la creación "en ese momento"

**Explorar objetos**

La herramienta SQL Developer cuenta con un explorador de objetos basado en árbol. Gracias a esto podemos acceder a todos los objetos que componen nuestra BDD de una manera fácil e intuitiva

**Crear objetos**

A la hora de crear objetos, SQL Developer nos facilita la tarea gracias a los diálogos/formularios de creación que tiene para cada tipo de objeto.

Entre los datos soportados incluye soporte para:

* Tablas externas.
* Tablas organizadas por índice.
* Tablas temporales.
* Tablas particionadas (Range, Hash y List)

**Modificar objetos**

SQL Developer a la hora de modificar objetos también nos ofrece unos diálogos específicos para facilitar la tarea. Entre dichos diálogos se encuentran:

Tabla -> Cambiar nombre. Para renombrar una tabla.

Columna -> Agregar Para añadir una columna a una tabla.

Vista -> Compilar. Para compilar una vista en busca de errores antes de su ejecución.

Indice -> Reconstruir. Para reconstruir un índice o alguna de sus particiones. Si un índice no es usable, una reconstrucción con éxito lo convierte en usable.

Procedimiento -> Debug. Poco que explicar aquí, permite debuggear un procedimiento.

Disparadores -> Desactivar. Para deshabilitar un disparador.

Secuencias -> Eliminar. Eliminar una secuencia.

**Consultar y actualizar tablas**

Otra de las funcionalidades disponible es la de consultar los datos de las tablas y actualizarlos. Los datos se nos presentan en una rejilla sobre la cual podemos realizar algunas operaciones de presentación tales como:

* Ordenación.
* Filtrado.
* Filas duplicadas.
* Vista de una sola tupla.

Sobre la rejilla con los datos podemos hacer las clásicas operaciones de inserción, modificación y borrado de tuplas de una manera muy intuitiva gracias a la interfaz que han desarrollado.

Tenemos más posibilidades: tiene soporte para CLOBs y BLOBs, existe un sistema de tracking de los cambios realizados sobre los datos mediante mensajes de log y cuenta con un generador de consultas visual para formar querys. Para activarlo, abrir una sesión de SQL, pulsar el botón derecho sobre la sesión y elegir "Generador de consultas"

**Exportar datos, DDLs e importar datos**

Otra de las características reseñables de SQL Developer es la capacidad que tiene para exportar e importar datos, así como DDLs.

Para hacer uso de estas funciones se emplea la función deseada en el menú contextual del objeto para el cual queramos exportar/importar. Por ejemplo, a la hora de consultar las opciones de exportación de una tabla de nuestra BD, las opciones disponibles son las siguientes:

Por el contrario, cuando queremos importar, la única opción disponible es la de importar desde una hoja de cálculo de Excel o fichero csv.

**B. Indique sus ventajas y desventajas sobre otras herramientas similares.**

Ventaja: sistema de gestión y control centralizado

Las sentencias de Oracle SQL permiten que los datos se controlen desde un repositorio central tabular. Un administrador de bases de datos (DBA por sus siglas en inglés) es responsable de crear usuarios, asignar privilegios, añadir registros, eliminar información redundante, modificar datos existentes y procesar preguntas. Estos datos almacenados centralmente son compartidos y accedidos por varias aplicaciones. Esto elimina la redundancia en la entrada y almacenamiento de datos.

Ventaja: estadarización

Una ventaja principal de Oracle SQL es su estandarización y consistencia entre distintas implementaciones. SQL fue estandarizado por primera vez por el ANSI (Instituto Estadounidense de Estandarización) en 1986, y luego ratificado en 1987 por la Organización Internacional de Estandarización (ISO), el cual sigue siendo el organismo de estandarización.

Desventaja: inhabilidad de implementar el procesamiento recursivo

Una de las mayores desventajas de SQL es su incapacidad de ejecutar procesamientos recursivos. El procesamiento recursivo es un tipo de función de computadora (o programa) en el cual uno de los pasos o procedimientos vuelve a hacer correr el programa entero (o el procedimiento). SQL carece de construcciones de tipo lazo que son comunes en otros tipos de lenguajes de programación de alto nivel. No se pueden repetir acciones y no hay forma de definir construcciones repetitivas en SQL.

Desventaja: incompatibilidad y complejidad

Una desventaja de Oracle SQL es la inconsistencia e incompatibilidad de datos en las áreas del tiempo y sintaxis de datos, concatenación de cadenas y sensibilidad de caracteres. El lenguaje es complejo, con un enfoque de palabras clave similar en estructura a COBOL (por las cifras en inglés de lenguaje común orientado a los negocios), con menos reglas de sintaxis y gramática.

Desventaja: funcionalidad limitada

SQL es un dominio específico o lenguaje de propósito especial, y su uso está limitado a un dominio de programa específico. Las sentencias de SQL son operadas en tablas y conjuntos de datos, como por ejemplo bases de datos de personal y hojas de cálculo de contabilidad. SQL es un lenguaje declarativo específico de dominio que está limitado a la representación tabular de los datos.

**Instalando**

**Instale la herramienta SQL Developer. ¿Son claras las instrucciones de instalación? ¿Se le presentó**

**algún problema?**

No son claras por que no sirve un link.

**Arrancando**

**Realice y explique cómo se deben realizar las siguientes acciones:**

**Establecer una conexión con el motor ORACLE de la ESCUELA**

Primero abrimos la opción de nueva conexión en dicho espacio ingresamos el nombre de la conexión, usuario, contraseña (en ambos casos es bd#carnet), nombre del Host (granate.is.escuelaing.edu.co), el número del puerto (1521) y el SID (ORCL).

**Consultar toda la información posible que hay en su cuenta**

**Bibliografia**

[**http://superinformacionweb.blogspot.com.co/2014/03/ventajas-y-desventajas-de-oracle.html**](http://superinformacionweb.blogspot.com.co/2014/03/ventajas-y-desventajas-de-oracle.html)

**http://www.v-espino.com/~chema/daw1/tutoriales/oracle/sqldeveloper.htm**