

Fundamentos de Bases de Datos.

Práctica 9.

Profesor: M.I. Gerardo Avilés Rosas

gar@ciencias.unam.mx

Laboratorio: Luis Eduardo Castro Omaña

lalo_castro@ciencias.unam.mx

18 de abril de 2017

Se dan a conocer especificaciones de entrega para la práctica 8.

1. Integridad de datos

1.1. Restricciones

Las restricciones le permiten definir la manera en que Motor de base de datos exigirá automáticamente la integridad de una base de datos. Las restricciones definen reglas relativas a los valores permitidos en las columnas y constituyen el mecanismo estándar para exigir la integridad. El uso de restricciones es preferible al uso de Desencadenadores DML¹, reglas y valores predeterminados. El optimizador de consultas también utiliza definiciones de restricciones para generar planes de ejecución de consultas de alto rendimiento.

SQL Server admite las siguientes clases de restricciones:

- NOT NULL especifica que la columna no acepta valores NULL. Un valor NULL no es lo mismo que cero (0), en blanco o que una cadena de caracteres de longitud cero, como . NULL significa que no hay ninguna entrada. La presencia de un valor NULL suele implicar que el valor es desconocido o no está definido. Se recomienda evitar la aceptación de valores NULL, dado que pueden implicar una mayor complejidad en las consultas y actualizaciones. Por otra parte, hay otras opciones para las columnas, como las restricciones PRIMARY KEY, que no pueden utilizarse con columnas que aceptan valores NULL.
- Las restricciones CHECK exigen la integridad del dominio mediante la limitación de los valores que se pueden asignar a una columna. Una restricción CHECK especifica una condición de búsqueda booleana (se evalúa

¹Se tratará con mas detalle en la siguiente práctica.

como TRUE, FALSE o desconocido) que se aplica a todos los valores que se indican en la columna. Se rechazan todos los valores que se evalúan como FALSE. En una misma columna se pueden especificar varias restricciones CHECK.

- Las restricciones UNIQUE exigen la unicidad de los valores de un conjunto de columnas. En una restricción UNIQUE, dos filas de la tabla no pueden tener el mismo valor en las columnas. Las claves principales también exigen exclusividad, pero no aceptan NULL como uno de los valores exclusivos.
- Las restricciones PRIMARY KEY identifican la columna o el conjunto de columnas cuyos valores identifican de forma exclusiva cada una de las filas de una tabla. Dos filas de la tabla no pueden tener el mismo valor de clave principal. No se pueden asignar valores NULL a ninguna de las columnas de una clave principal. Se recomienda utilizar una columna pequeña de tipo entero como clave principal. Todas las tablas tienen que tener una clave principal. Una columna o combinación de columnas certificada como valor de clave principal se denomina clave candidata.
- Las restricciones FOREIGN KEY identifican y exigen las relaciones entre las tablas. Una clave externa de una tabla apunta a una clave candidata de otra tabla. No se puede insertar una fila que tenga un valor de clave externa, excepto NULL, si no hay una clave candidata con dicho valor. La cláusula ON DELETE controla las acciones que se llevarán a cabo si intenta eliminar una fila a la que apuntan las claves externas existentes. La cláusula ON DELETE tiene las siguientes opciones:
 - NO ACTION especifica que la eliminación produce un error.
 - CASCADE especifica que también se eliminan todas las filas con claves externas que apuntan a la fila eliminada.
 - SET NULL especifica que todas las filas con claves externas que apuntan a la fila eliminada se establecen en NULL.
 - SET DEFAULT especifica que todas las filas con claves externas que apuntan a la fila eliminada se establecen en sus valores predeterminados.

La cláusula ON UPDATE define las acciones que se llevarán a cabo si intenta actualizar un valor de clave candidata a la que apuntan las claves externas existentes. Esta cláusula también admite las opciones NO ACTION, CASCADE, SET NULL y SET DEFAULT.

1.2. Incrementadores

1.2.1. IDENTITY

Cada tabla debe contener una columna como llave primaria. Es ampliamente recomendable utilizar una columna que se incremente por cada registro en la

tabla, dicha columna podría ser la llave primaria o parte de ella de la tabla que la contenga. En Transact-SQL se crea una columna de identidad en una tabla; la cual, genera un campo que se incrementara en *i* cada vez que se realice un registro en la tabla que la contiene. Esta propiedad se usa con las instrucciones CREATE TABLE y ALTER TABLE de Transact-SQL, se declara como sigue:

```
IDENTITY [ (valor_inicio , incremento) ]
```

- **valor_inicio:** Es el valor que se utiliza para la primera fila cargada en la tabla.
- **incremento:** Se trata del valor incremental que se agrega al valor de identidad de la anterior fila cargada.

Debe especificar tanto el valor de inicialización como el incremento, o bien ninguno de los dos. Si no se especifica ninguno, el valor predeterminado es (1,1).

1.2.2. Secuencias

Una secuencia es un objeto enlazado a un esquema definido por el usuario que genera una secuencia de valores numéricos según la especificación con la que se creó la secuencia. La secuencia de valores numéricos se genera en orden ascendente o descendente en un intervalo definido y se puede configurar para reiniciarse (en un ciclo) cuando se agota. Las secuencias, a diferencia de las columnas de identidad, no se asocian a tablas concretas. Las aplicaciones hacen referencia a un objeto de secuencia para recuperar su valor siguiente. La aplicación controla la relación entre las secuencias y tablas. Las aplicaciones de usuario pueden hacer referencia un objeto de secuencia y coordinar los valores a través de varias filas y tablas.

A diferencia de los valores de las columnas de identidad que se generan cuando se insertan filas, una aplicación puede obtener el número de secuencia siguiente sin insertar la fila mediante una llamada a la función NEXT VALUE FOR.

```
CREATE SEQUENCE [nombre_esquema . ] nombre_secuencia
[ AS [ tipo_dato ] ]
[ START WITH <i> ]
[ INCREMENT BY <i> ]
[ { MINVALUE [ <i> ] } | { NO MINVALUE } ]
[ { MAXVALUE [ <i> ] } | { NO MAXVALUE } ]
[ CYCLE | { NO CYCLE } ]
[ { CACHE [ <i> ] } | { NO CACHE } ]
[ ; ]
```

Los números de secuencia se generan fuera del ámbito de la transacción actual. Se utilizan tanto si la transacción que usa el número de secuencia se confirma como si se revierte.

2. SQL Server Management Studio

SQL Server Management Studio (SSMS) es un entorno integrado para acceder, configurar, administrar y desarrollar todos los componentes de SQL Server. SSMS combina un amplio grupo de herramientas gráficas con una serie de editores de scripts enriquecidos que permiten a los desarrolladores y administradores de todos los niveles acceder a SQL Server.

3. Diccionario de datos

Contiene las características lógicas de los sitios donde se almacenan los datos del sistema, incluyendo nombre, descripción, alias, contenido y organización. Identifica los procesos donde se emplean los datos y los sitios donde se necesita el acceso inmediato a la información, se desarrolla durante el análisis de flujo de datos y auxilia a los analistas que participan en la determinación de los requerimientos del sistema, su contenido también se emplea durante el diseño.

Cada entrada en el diccionario consiste de un conjunto de detalles que describen los datos utilizados o producidos por el sistema. Cada uno está identificado con:

- Un nombre: para distinguir un dato de otro.
- Descripción: indica lo que representa en el sistema.
- Alias: porque un dato puede recibir varios nombres, dependiendo de quien uso este dato.
- Longitud: porque es de importancia de saber la cantidad de espacio necesario para cada dato.
- Valores de los datos: porque en algunos procesos solo son permitidos valores muy específicos para los datos. Si los valores de los datos están restringidos a un intervalo específico, esto debe estar en la entrada del diccionario.

La propiedad de identidad de una columna no garantiza lo siguiente:

- Unicidad del valor - exclusividad debe exigirse a través de un PRIMARY KEY o UNIQUE restricción o UNIQUE índice.
- Valores consecutivos dentro de una transacción : no se garantiza que una transacción insertar varias filas obtenga valores consecutivos para las filas, ya que podría producirse otras inserciones simultáneas en la tabla. Si los valores deben ser consecutivos, a continuación, la transacción debe usar un bloqueo exclusivo en la tabla o usar el SERIALIZABLE nivel de aislamiento.

- Valores consecutivos después de reiniciar el servidor u otros errores –SQL Server podría guardar en caché los valores de identidad por motivos de rendimiento y algunos de los valores asignados podrían perderse durante un reinicio del servidor o error de base de datos. Esto puede tener como resultado espacios en el valor de identidad al insertarlo. Si no es aceptable que haya espacios, la aplicación debe usar mecanismos propios para generar valores de clave. Mediante un generador de secuencias con el NO-CACHE opción puede limitar los espacios a transacciones que nunca se hayan confirmado.
- Reutilización de valores : para una propiedad de identidad especificada con el valor de inicialización e incremento específico, la identidad que el motor no reutiliza los valores. Si una instrucción de inserción concreta produce un error o si la instrucción de inserción se revierte, los valores de identidad utilizados se pierden y no volverán a generarse. Esto puede tener como resultado espacios cuando se generan los valores de identidad siguientes.

4. Actividad

Deberán descargar e instalar SQL Server Management Studio de la siguiente dirección: <https://go.microsoft.com/fwlink/?LinkID=840946>. Una vez que tengan instalado SQL Server Management Studio deberán correr los scripts DDL.sql que crearon en la práctica pasada pero en una nueva base de datos.

En la nueva base de datos que crearon deberán completar el esquema de la base de datos de la asociación. Cada una de las tablas deberá tener las restricciones correspondientes, así como sus llaves primarias y las llaves foraneas de las tablas con las que se encuentre relacionada.

Por ultimo, deberán generar un diccionario de datos de la base de datos.

5. Entregables

Se debe crear un script DDL.sql el cual contendrá las instrucciones para la creación de la base de datos, sus tablas y sus restricciones.

El diccionario de datos deberá ser generado en un documento pdf, el cual lo deberán guardar dentro de la carpeta DOC.

La entrega deberá ser el día lunes 24 de Abril de 2017.