

Uso de las herramientas de consulta de Transact-SQL

Contenido

Introducción	1
Analizador de consultas SQL	2
Uso de la herramienta Examinador de objetos en el Analizador de consultas SQL	3
Uso de plantillas en el Analizador de consultas SQL	5
Uso de la utilidad osql	6
Ejecución de instrucciones de Transact-SQL	9

Notas para el instructor

Microsoft® SQL Server™ 2000 proporciona varias herramientas de consulta que pueden usarse para ejecutar secuencias de comandos de Transact-SQL. Este módulo describe cómo se puede usar el Analizador de consultas SQL y la utilidad de línea de comandos **osql**, además de mostrar varias formas en que se pueden ejecutar las instrucciones de Transact-SQL.

Al terminar este módulo, los alumnos serán capaces de:

- Describir las funciones básicas del Analizador de consultas SQL.
- Describir cómo se usa la herramienta Examinador de objetos en el Analizador de consultas SQL.
- Describir cómo se usan las plantillas en el Analizador de consultas SQL.
- Describir cómo se usa la utilidad de línea de comandos **osql**.
- Ejecutar de varias formas instrucciones de Transact-SQL.

Introducción

Objetivos de la diapositiva

Proporcionar una introducción a los temas y objetivos del módulo.

Explicación previa

En este módulo aprenderá acerca de algunas de las herramientas de consulta que proporciona SQL Server para la ejecución de secuencias de comandos de Transact-SQL.

- Analizador de consultas SQL
- Uso de la herramienta Examinador de objetos en el Analizador de consultas SQL
- Uso de plantillas en el Analizador de consultas SQL
- Uso de la utilidad osql
- Ejecución de instrucciones de Transact-SQL

Microsoft® SQL Server™ 2000 proporciona varias herramientas de consulta que pueden usarse para ejecutar secuencias de comandos de Transact-SQL. Este módulo describe cómo se puede usar el Analizador de consultas SQL y la utilidad de línea de comandos **osql**, además de mostrar varias formas en que se pueden ejecutar las instrucciones de Transact-SQL.

Al terminar este módulo, el alumno será capaz de:

- Describir las funciones básicas del Analizador de consultas SQL.
- Describir cómo se usa la herramienta Examinador de objetos en el Analizador de consultas SQL.
- Describir cómo se usan las plantillas en el Analizador de consultas SQL.
- Describir cómo se usa la utilidad de línea de comandos **osql**.
- Ejecutar de varias formas instrucciones de Transact-SQL.

Analizador de consultas SQL

Objetivos de la diapositiva

Presentar la herramienta Analizador de consultas SQL.

Explicación previa

El Analizador de consultas SQL se usa en SQL Server para ver de forma gráfica instrucciones de consulta y resultados.

- Un procesador de texto sin formato
- Codificación en color
- Múltiples ventanas de consulta con salida de texto o en cuadrícula
- Ejecución parcial de secuencias de comandos
- Información de ejecución de la consulta

Sugerencia

Haga una demostración del Analizador de consultas de SQL Server. Muestre a los alumnos los elementos básicos de la ventana Analizador de consultas de SQL Server, incluidas las tres formas de ejecutar una consulta, la codificación de colores de la sintaxis y cómo abrir y guardar una secuencia de comandos.

El Analizador de consultas SQL se usa en SQL Server para ver de forma gráfica instrucciones de consulta y resultados. También puede utilizarse para escribir, modificar y guardar secuencias de comandos de Transact-SQL.

El Analizador de consultas SQL proporciona además otras herramientas que permiten determinar de qué forma SQL Server está interpretando y ejecutando una instrucción de Transact-SQL.

El Analizador de consultas SQL incluye:

- *Un procesador de texto sin formato.* Este procesador incluye funciones avanzadas de procesamiento de textos como es el uso de sangrías de bloques, comentarios de bloque o eliminación de comentarios, y la conversión a mayúsculas o minúsculas.
- *Codificación en color.* Cuando escribe una consulta, el Analizador de consultas de SQL Server resalta las palabras clave, las cadenas de caracteres y otros elementos del lenguaje; se puede personalizar la forma en que aparecen por medio de la codificación en color.
- *Múltiples ventanas de consulta con salida de texto o en cuadrícula.* Cada ventana de consulta tiene su propia conexión con un servidor SQL Server. Los resultados se pueden ver en una ventana de texto o en una cuadrícula
- *Ejecución parcial de secuencias de comandos.* Esta posibilidad permite ejecutar partes de una secuencia de comandos. Puede seleccionar partes de una secuencia de comandos y SQL Server ejecutará sólo esas partes.
- *Información de ejecución de la consulta.* La información de ejecución de la consulta incluye aspectos tales como datos estadísticos del cliente, información de seguimiento del servidor y datos del plan de ejecución. Puede usar esa información como ayuda para ajustar y solucionar problemas de las secuencias de comandos.

Uso de la herramienta Examinador de objetos en el Analizador de consultas SQL

Objetivos de la diapositiva

Presentar la herramienta Examinador de objetos en el Analizador de consultas SQL.

Explicación previa

Puede usar la herramienta Examinador de objetos para desplazarse por las tres vistas de los objetos en una base de datos y poder llegar así a un objeto determinado.

- El Examinador de objetos habilita el desplazamiento por las tres vistas de los objetos en una base de datos
- Con el Examinador de objetos puede:
 - Crear secuencias de comandos para los objetos
 - Ejecutar procedimientos almacenados
 - Abrir tablas
 - Modificar los objetos de la base de datos
 - Usar plantillas de Transact-SQL

Puede usar la herramienta Examinador de objetos para desplazarse por las tres vistas de los objetos en una base de datos y poder llegar así a un objeto determinado. Esta herramienta también crea secuencias de comandos para los objetos, ejecuta procedimientos almacenados y permite tener acceso a tablas y vistas.

Con el Examinador de objetos puede:

- Crear secuencias de comandos para los objetos.

Las operaciones que se pueden realizar dependen del tipo de objeto. Por ejemplo, para los objetos de tabla se pueden generar secuencias de comandos que contienen instrucciones SELECT, instrucciones de definición de datos como CREATE o instrucciones de tratamiento de datos como INSERT.

- Ejecutar procedimientos almacenados.

Al ejecutar un procedimiento almacenado que tiene un parámetro, el Examinador de objetos pide que se le dé un valor.

- Abrir tablas.

El Examinador de objetos muestra los resultados de la consulta de forma independiente. Puede modificar, insertar o eliminar filas.

- Modificar los objetos de la base de datos.

Puede ver y modificar los objetos de la base de datos. El Examinador de objetos muestra, en el panel Editor, una instrucción ALTER para el objeto seleccionado. Por ejemplo, si el objeto seleccionado es un procedimiento almacenado, el Examinador de objetos proporciona una instrucción ALTER PROCEDURE. Puede usar esa instrucción ALTER para especificar los cambios y, a continuación, ejecutarla.

- Usar plantillas de Transact-SQL.

Estas plantillas contienen secuencias de comandos de Transact-SQL que le ayudan a crear objetos en una base de datos. Puede usar estas plantillas para:

- Crear bases de datos, tablas, vistas, índices, procedimientos almacenados, desencadenadores, estadísticas y funciones.
- Administrar propiedades ampliadas, servidores vinculados, cuentas de inicio de sesión, funciones y usuarios.
- Declarar y usar cursores.
- Personalizar secuencias de comandos.

Uso de plantillas en el Analizador de consultas SQL

Objetivos de la diapositiva
Describir cómo se usan las plantillas en el Analizador de consultas SQL.

Explicación previa
El Analizador de consultas SQL incluye plantillas que se pueden usar como punto de partida en la creación de objetos en una base de datos.

- **Plantillas**
 - Se usan como punto de partida en la creación de objetos en una base de datos
 - Contienen parámetros que le ayudarán a personalizar las secuencias de comandos
- **Formato para definiciones de parámetros de plantillas**
<nombreParámetro, tipoDatos, valor>

Sugerencia
Describa la finalidad y forma de uso de las plantillas incluidas junto con el Analizador de consultas SQL.

Demuestre de forma breve cómo se puede tener acceso a esas plantillas y muestre cómo funciona la interfaz gráfica.

El Analizador de consultas SQL incluye plantillas que se pueden usar como punto de partida en la creación de objetos en una base de datos.

SQL Server proporciona diversas plantillas dentro del directorio Templates\SQL Query Analyzer. Entre ellas se encuentran plantillas que crean bases de datos, tablas, vistas, índices, procedimientos almacenados, desencadenadores, estadísticas y funciones. Otras plantillas de ese directorio sirven de ayuda para administrar propiedades ampliadas, servidores vinculados, inicios de sesión, funciones y usuarios, además de para declarar y usar cursores.

Las secuencias de comandos de las plantillas que se incluyen en el Analizador de consultas SQL contienen parámetros que le ayudarán a personalizar las secuencias de comandos. Las definiciones de parámetros de las plantillas tienen este formato:

<nombreParámetro, tipoDatos, valor>

La tabla siguiente describe las definiciones y formato de los parámetros de plantillas.

Formato	Definición de parámetro de plantillas
<nombreParámetro>	Nombre del parámetro en la secuencia de comandos
<tipoDatos>	Tipo de datos del parámetro
<valor>	Valor que debe sustituir al parámetro en todos los lugares en que aparece en la secuencia de comandos

Para insertar valores en la secuencia de comandos se usa un cuadro de diálogo. Por ejemplo, al ejecutar una función desde el Examinador de objetos, la función que aparece en el panel de edición contiene todas las definiciones de parámetro de los argumentos de la función. Así, puede usar el cuadro de diálogo Reemplazar parámetros de plantilla para especificar los valores específicos del argumento.

Uso de la utilidad osql

Objetivos de la diapositiva

Presentar la utilidad **osql**.

Explicación previa

La utilidad **osql** se ejecuta desde la línea de comandos y permite realizar consultas en SQL Server

- Inicio de la utilidad de línea de comandos **osql**
- Uso del modo interactivo
- Uso del modo de ejecución de secuencias de comandos
- Uso de comandos ampliados en secuencias de comandos **osql**

Para su información

La herramienta **isql** no se describe en este curso ya que utiliza bibliotecas de bases de datos para comunicarse con el servidor y no admite los tipos de datos Unicode.

La utilidad **osql** se ejecuta desde la línea de comandos para ejecutar de forma interactiva instrucciones y secuencias de comandos de Transact-SQL. Para usar la utilidad los usuarios deben conocer Transact-SQL y saber cómo se ejecutan secuencias de comandos desde el símbolo del sistema.

La utilidad de línea de comandos **osql** usa Conectividad abierta de bases de datos (ODBC, *Open Database Connectivity*) de SQL Server para comunicarse con el servidor. Por eso, está sometida a las limitaciones y peculiaridades de uso de la interfaz de programación de aplicaciones (API) de ODBC.

Inicio de la utilidad de línea de comandos osql

La utilidad de línea de comandos **osql** se inicia directamente desde el sistema operativo con las opciones que se enumeran a continuación, en las que se distinguen mayúsculas y minúsculas. Se puede llamar también desde un archivo de proceso por lotes o desde el símbolo del sistema. Un *lote* es un conjunto de instrucciones de Transact-SQL que se emiten juntas y se ejecutan como un grupo.

Uso del modo interactivo

Una vez iniciado, la utilidad de línea de comandos **osql** acepta instrucciones de Transact-SQL y las envía a SQL Server de forma interactiva. Los resultados reciben el formato adecuado y se muestran en la pantalla.

Use la instrucción **GO** para ejecutar las instrucciones de Transact-SQL que se encuentran en el búfer de entrada. Use la instrucción **QUIT** o **EXIT** para salir de la utilidad de línea de comandos **osql**.

Uso del modo de ejecución de secuencias de comandos

Los usuarios emiten un proceso por lotes **osql** que especifica una única instrucción de Transact-SQL que ejecuta o dirige la utilidad a un archivo de texto que contiene las instrucciones de Transact-SQL que se deben ejecutar. La salida se suele dirigir a un archivo de texto, aunque también se puede mostrar en la ventana del símbolo del sistema.

Sintaxis parcial

osql [-S *nombreServidor*] [-E] [-U *idInicioSesión*] [-P *contraseña*]
[-i *archivoEntrada*] [-o *archivoSalida*] [-?]

Nota En los parámetros de las instrucciones **osql** se distinguen las mayúsculas y las minúsculas.

Sugerencia

Mencione que los argumentos pueden ir precedidos de guiones (-) o barras diagonales (/).

Para obtener más información acerca de los argumentos, indique a los alumnos que pueden consultar los Libros en pantalla de SQL Server y buscar el tema acerca de la utilidad **osql**.

Recuerde a los alumnos que los argumentos pueden ir precedidos de guiones (-) o barras diagonales (/). La tabla siguiente describe los argumentos más utilizados.

Argumento	Descripción
-S <i>nombreServidor</i>	Especifica el servidor SQL Server al que conectarse. <i>nombreServidor</i> es el nombre del equipo servidor en la red. Esta opción se requiere si se ejecuta osql desde un equipo remoto de la red.
-E	Utiliza una conexión de confianza en lugar de solicitar una contraseña.
-U <i>IdInicioSesión</i>	Es el identificador de inicio de sesión del usuario. En los identificadores de inicio de sesión se distinguen las mayúsculas y las minúsculas. Si no se utiliza la opción -U o -P , SQL Server utiliza la cuenta de usuario de quien esté conectado actualmente, con lo que no pedirá contraseña.
-P <i>contraseña</i>	Es una contraseña especificada por el usuario. Si no se utiliza la opción -P , osql pide una contraseña. Si se utiliza la opción -P al final del símbolo del sistema sin ninguna contraseña, osql utiliza la contraseña predeterminada (NULL). En las contraseñas se distinguen mayúsculas y minúsculas. Si no se utiliza la opción -U o -P , SQL Server utiliza la cuenta de usuario de quien esté conectado actualmente, con lo que no pedirá contraseña.
-i <i>archivoEntrada</i>	Identifica el archivo que contiene un lote de instrucciones de Transact-SQL o de procedimientos almacenados. En lugar de -i se puede utilizar el símbolo menor que (<).
-o <i>archivoSalida</i>	Identifica el archivo que recibe el resultado de osql . En lugar de -o se puede utilizar el símbolo mayor que (>). Si el archivo de entrada es Unicode, el archivo de salida será Unicode si se especifica -o . Si el archivo de entrada no es Unicode, el archivo de salida será OEM.
-?	Muestra el resumen de la sintaxis de los modificadores de osql .

Uso de comandos ampliados en secuencias de comandos **osql**

La utilidad de línea de comandos **osql** puede procesar también comandos que no sean instrucciones de Transact-SQL. La herramienta sólo reconoce esos comandos cuando se encuentran al principio de la línea o inmediatamente detrás del símbolo del sistema de **osql**. Todas las instrucciones que puedan aparecer a continuación en esa misma línea se omitirán.

La tabla siguiente describe estos comandos adicionales.

Comando	Descripción
GO	Ejecuta todos los comandos que se han escrito desde el comando GO anterior
RESET	Borra las instrucciones que se hayan escrito anteriormente
ED	Efectúa una llamada al procesador de textos
!! <i>comando</i>	Ejecuta un comando del sistema operativo
QUIT o EXIT()	Sale del programa osql
CTRL+C	Finaliza una consulta sin salir de osql

◆ Ejecución de instrucciones de Transact-SQL

Objetivos de la diapositiva

Proporcionar una introducción a las formas de ejecutar instrucciones de Transact-SQL.

Explicación previa

Puede ejecutar instrucciones de Transact-SQL de varias formas.

- Construir instrucciones dinámicamente
- Utilizar lotes
- Uso de secuencias de comandos

Puede ejecutar instrucciones de Transact-SQL de varias formas:

- Puede construir dinámicamente instrucciones en tiempo de ejecución.
- Puede usar lotes para agrupar instrucciones que se deben ejecutar conjuntamente.
- Puede usar secuencias de comandos para guardar lotes en un archivo de forma que se puedan volver a usar en el futuro.

Construir instrucciones dinámicamente

Objetivos de la diapositiva

Presentar a los alumnos la ejecución dinámica de las instrucciones.

Explicación previa

Se pueden generar instrucciones dinámicamente, con lo que se crean al mismo tiempo que se ejecuta la secuencia de comandos

- Utilice EXECUTE con literales de cadena y variables
- Utilícela cuando tenga que asignar el valor de la variable en tiempo de ejecución

Ejemplo 1

```
USE library
DECLARE @dbname varchar(30), @tablename varchar(30)
SET @dbname = 'northwind'
SET @tablename = 'products'

EXECUTE
('USE ' + @dbname + ' SELECT productname FROM ' +
@tablename)
GO
```

Se pueden generar instrucciones dinámicamente, con lo que se crean al mismo tiempo que se ejecuta la secuencia de comandos.

Para crear una instrucción dinámicamente, utilice la instrucción EXECUTE con un conjunto de literales de cadena y variables que se resuelven en tiempo de ejecución.

Las instrucciones creadas dinámicamente son útiles cuando desea que el valor de la variable se asigne al ejecutarse la instrucción. Por ejemplo, puede crear una instrucción dinámica que ejecute la misma acción sobre un conjunto de objetos de la base de datos.

Sintaxis

```
EXECUTE ({@varCadena | 'cadenaTsql'} + [{@varCadena |
'cadenaTsql'}...])
```

Tenga en cuenta los siguientes hechos acerca de la instrucción EXECUTE:

- La instrucción EXECUTE ejecuta instrucciones compuestas de cadenas de caracteres en un lote de Transact-SQL. Como se trata de literales de cadena, asegúrese de que agrega espacios en los lugares adecuados para garantizar una concatenación adecuada.
- La instrucción EXECUTE puede incluir un literal de cadena, una variable local de cadena o una concatenación de ambas.
- Todos los elementos de la cadena EXECUTE deben constar de datos de caracteres; debe convertir todos los datos numéricos antes de utilizar la instrucción EXECUTE.
- No puede usar funciones para generar la cadena de texto para la ejecución en la instrucción EXECUTE.
- Puede crear dinámicamente cualquier instrucción de Transact-SQL válida, incluidas las funciones.
- Las instrucciones EXECUTE se pueden anidar.

- Las variables y las tablas temporales que se crean dinámicamente duran sólo el tiempo necesario para que se ejecute la instrucción.

Ejemplo 1

Este ejemplo muestra cómo se utiliza una instrucción ejecutada dinámicamente para especificar un contexto de base de datos distinto de aquél en el que se encuentra en la actualidad y para seleccionar todas las columnas y filas de una tabla especificada. En este ejemplo, el cambio del contexto de la base de datos a **northwind** sólo se prolonga durante el tiempo que dure la consulta. El contexto de la base de datos actual no se cambia.

Mediante un procedimiento almacenado, el usuario podría pasar la información de la base de datos y de la tabla a la instrucción como parámetros y, después, consultar una tabla específica de una base de datos.

```
USE library
DECLARE @dbname varchar(30), @tablename varchar(30)
SET @dbname = 'northwind'
SET @tablename = 'products'

EXECUTE
    ('USE ' + @dbname + ' SELECT productname FROM ' +
    @tablename)
GO
```

Resultado

<u>productname</u>
Chai
Chang
Aniseed Syrup

Ejemplo 2

Este ejemplo muestra cómo se puede utilizar una instrucción ejecutada dinámicamente para cambiar una opción de base de datos durante la ejecución de la instrucción. La instrucción siguiente no devuelve un contador del número de filas afectadas.

```
USE northwind
EXECUTE ('SET NOCOUNT ON '+ 'SELECT lastname, reportsto FROM
employees WHERE reportsto IS NULL')
GO
```

Resultado

<u>lastname</u>	<u>reportsto</u>
Fuller	NULL

Utilizar lotes

Objetivos de la diapositiva

Explicar a los alumnos el uso de los lotes.

Explicación previa

Un lote es un conjunto de instrucciones de Transact-SQL que se emiten juntas y se ejecutan como un grupo.

- Una o varias instrucciones Transact-SQL emitidas juntas
- Definición de un lote con la instrucción GO
- Cómo procesa SQL Server los lotes
- Instrucciones que no se pueden combinar en un lote
 - CREATE PROCEDURE
 - CREATE VIEW
 - CREATE TRIGGER
 - CREATE RULE
 - CREATE DEFAULT

En un lote se puede emitir una o varias instrucciones.

Una o varias instrucciones Transact-SQL emitidas juntas

Un *lote* es un conjunto de instrucciones de Transact-SQL que se emiten juntas y se ejecutan como un grupo. Los lotes se pueden ejecutar interactivamente o como parte de una secuencia de comandos. Una secuencia de comandos puede incluir más de un lote de instrucciones de Transact-SQL.

Definición de un lote con la instrucción GO

Utilice una instrucción GO para señalar el final de un lote. GO no es una instrucción de Transact-SQL aceptada universalmente; sólo la admiten el Analizador de consultas SQL y la utilidad **osql**. Las aplicaciones basadas en las interfaces de programación de aplicaciones (API) ODBC u OLE DB generan un error de sintaxis cuando tratan de ejecutar una instrucción GO.

Cómo procesa SQL Server los lotes

SQL Server optimiza, compila y ejecuta juntas las instrucciones de un lote. Sin embargo, las instrucciones no tienen que ejecutarse necesariamente como una unidad de trabajo recuperable.

El ámbito de las variables definidas por el usuario se limita a un lote, con lo que no se puede hacer referencia a una variable después de una instrucción GO.

Nota Si existe un error de sintaxis en un lote, no se ejecutará ninguna de las instrucciones de ese lote. La ejecución se inicia con el siguiente lote.

Sugerencia

Describe y compare cada uno de los ejemplos.

Instrucciones que no se pueden combinar en un lote

Determinadas instrucciones de creación de objetos se deben ejecutar en sus propios lotes, dentro de una secuencia de comandos, dada la forma en que aquéllas se definen. Estas instrucciones de creación de objetos aparecen identificadas por medio de una estructura peculiar: un encabezado de definición de objeto seguido de la palabra clave AS y una o varias instrucciones de definición, que se finaliza con un comando GO.

Estas instrucciones se deben ejecutar en lotes independientes.

- CREATE PROCEDURE
- CREATE VIEW
- CREATE TRIGGER
- CREATE RULE
- CREATE DEFAULT

Ejemplo 1

Este ejemplo muestra instrucciones que no se ejecutan correctamente en un único lote, ya que la consulta combina de forma errónea instrucciones que no pueden combinarse en un mismo lote. Debe insertar una instrucción GO antes de cada una de las instrucciones CREATE VIEW para que se ejecuten correctamente.

```
CREATE DATABASE ...  
CREATE TABLE ...  
CREATE VIEW ...  
CREATE VIEW ...  
GO
```

Ejemplo 2

Este ejemplo agrupa las instrucciones usadas en el ejemplo 1 en los lotes correspondientes para que se puedan ejecutar correctamente.

```
CREATE DATABASE ...  
CREATE TABLE ...  
GO
```

```
CREATE VIEW ...  
GO
```

```
CREATE VIEW ...  
GO
```

Uso de secuencias de comandos

Objetivos de la diapositiva

Presentar cómo se utilizan las secuencias de comandos para ejecutar instrucciones de Transact-SQL.

Explicación previa

Las secuencias de comandos son una de las formas más comunes de ejecutar instrucciones de Transact-SQL.

- Una secuencia de comandos es una o más instrucciones Transact-SQL guardadas como un archivo con la extensión .sql
- Las secuencias de comandos:
 - contienen instrucciones guardadas
 - pueden escribirse utilizando cualquier procesador de texto
 - pueden volver a crear objetos de bases de datos o ejecutar instrucciones de forma repetida
 - pueden ejecutarse en el Analizador de consultas SQL o con la utilidad **osql**

Las *secuencias de comandos* son una de las formas más comunes de ejecutar instrucciones de Transact-SQL. Se componen de una o varias instrucciones de Transact-SQL que se guardan como un archivo.

Las secuencias de comandos se pueden escribir y guardar en el Analizador de consultas SQL o en cualquier procesador de texto, como Bloc de notas. El archivo de comandos se guarda con la extensión de nombre de archivo .sql.

Las secuencias de comandos guardadas son muy útiles para volver a crear bases de datos u objetos de datos, o cuando se debe utilizar un conjunto de instrucciones de forma repetida.

Puede abrir y ejecutar el archivo de comandos en el Analizador de consultas SQL o con la utilidad **osql** (u otra herramienta de consulta).