

Pregrado

Programa de
Ingeniería de
Sistemas

**GESTIÓN DE DATOS
E
INFORMACIÓN II**

Sesión 11

Tema:

Merge





Resultado de aprendizaje

Implementa procedimientos almacenados, cursores y triggers sobre una base de datos utilizando T-SQL para resolver las diversas actividades de una organización.

Evidencia de aprendizaje

Los estudiantes demostrarán su comprensión y habilidades en el uso de la declaración MERGE en SQL Server que permita realizar operaciones INSERT, UPDATE y DELETE en una tabla de destino basándose en una condición de coincidencia.

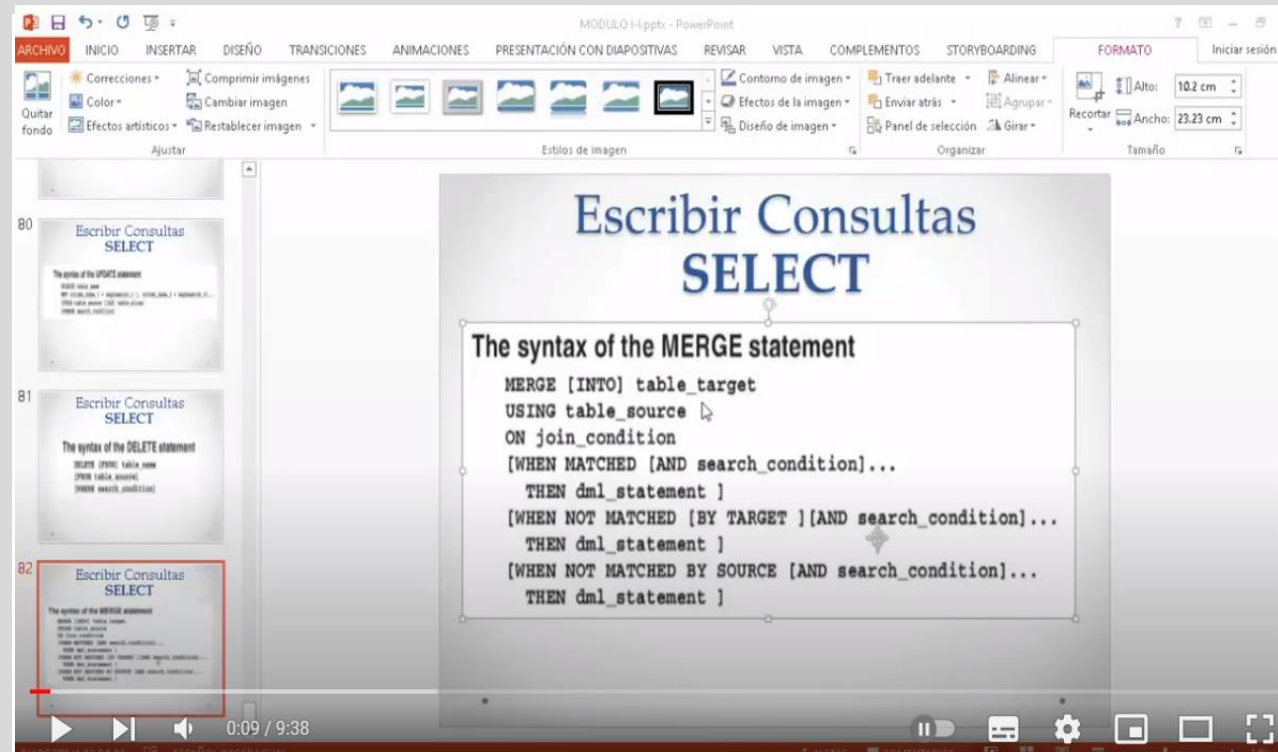


Contenido

Nombre del tema

- Merge

Revisa el siguiente video:



<https://youtu.be/hE4d5a8xmMg?si=6QRekS4JgnZvUhLQ>

Después de haber visualizado el video en la slide anterior, reflexionamos y respondemos las siguientes interrogantes:

01

¿Para qué sirve los Merge?

02

¿En que situaciones puedo utilizar Merge?

03

¿Qué ventajas ofrece utilizar Merge?





Merge

Tema

MERGE

Es una herramienta poderosa para realizar operaciones de inserción, actualización y eliminación de manera eficiente, pero su uso debe considerarse cuidadosamente en función de las características específicas del sistema y los requisitos de la aplicación.

MERGE: VENTAJAS

Eficiencia: MERGE permite realizar operaciones de inserción, actualización y eliminación en una sola sentencia, lo que puede ser más eficiente y reducir la cantidad de interacciones con la base de datos.

Atomicidad: La operación MERGE es atómica, lo que significa que todas las acciones (inserción, actualización, eliminación) se realizan en conjunto o ninguna se realiza. Esto garantiza la consistencia de los datos.

MERGE: VENTAJAS

Legibilidad del Código: La sentencia MERGE proporciona una forma más legible y concisa de expresar operaciones complejas que involucran varias acciones en comparación con usar sentencias separadas de INSERT, UPDATE y DELETE.

Menos bloqueo: Puede haber menos bloqueo de recursos durante la ejecución de una sentencia MERGE en comparación con la ejecución de sentencias individuales de INSERT, UPDATE y DELETE.

MERGE: DESVENTAJAS

Complejidad: La sintaxis de la sentencia MERGE puede ser compleja y puede ser más difícil de entender y mantener, especialmente para usuarios menos experimentados.

Problemas de Rendimiento: En algunas situaciones, el rendimiento de MERGE puede no ser tan eficiente como realizar operaciones individuales de INSERT, UPDATE y DELETE, especialmente en grandes conjuntos de datos.

MERGE: DESVENTAJAS

Problemas de Concurrencia: En entornos de alta concurrencia, MERGE puede encontrarse con problemas de bloqueo, y puede requerir ajustes en la configuración de aislamiento para manejar situaciones concurrentes.

Limitaciones de la Base de Datos: Algunas bases de datos pueden tener limitaciones en la implementación de la sentencia MERGE. Además, su comportamiento puede variar entre diferentes sistemas de gestión de bases de datos.

MERGE: EJEMPLO

Supongamos que tienes dos tablas: **TablaDestino** y **TablaOrigen**. Quieres sincronizar los datos de la **TablaOrigen** con la **TablaDestino**. La lógica sería insertar nuevos registros, actualizar los existentes y eliminar los que ya no están en la **TablaOrigen**.

Puedes lograr esto con **MERGE**. Ejemplo:

```
-- Crear las tablas de ejemplo
CREATE TABLE TablaDestino (
    ID INT PRIMARY KEY,
    Nombre VARCHAR(50),
    Edad INT
);

CREATE TABLE TablaOrigen (
    ID INT PRIMARY KEY,
    Nombre VARCHAR(50),
    Edad INT
);
```

```
-- Insertar datos de ejemplo
INSERT INTO TablaDestino VALUES (1, 'Juan', 25);
INSERT INTO TablaDestino VALUES (2, 'María', 30);
INSERT INTO TablaDestino VALUES (3, 'Carlos', 22);

INSERT INTO TablaOrigen VALUES (1, 'Juan', 26);
INSERT INTO TablaOrigen VALUES (2, 'María', 31);
INSERT INTO TablaOrigen VALUES (4, 'Laura', 28);
```


MERGE: EJEMPLO

```
-- Utilizar la operación MERGE para sincronizar las tablas
MERGE INTO TablaDestino AS destino
USING TablaOrigen AS origen
ON destino.ID = origen.ID
-- Cuando la coincidencia existe, actualiza la información
WHEN MATCHED THEN
    UPDATE SET destino.Nombre = origen.Nombre, destino.Edad = origen.Edad
-- Cuando no hay coincidencia, inserta un nuevo registro
WHEN NOT MATCHED BY TARGET THEN
    INSERT (ID, Nombre, Edad) VALUES (origen.ID, origen.Nombre, origen.Edad)
-- Cuando hay registros en la TablaDestino que no están en la TablaOrigen, elimina es
WHEN NOT MATCHED BY SOURCE THEN
    DELETE;

-- Ver los resultados en la TablaDestino
SELECT * FROM TablaDestino;
```

MERGE

MATCHED indica que hay una coincidencia entre las tablas fuente y destino.

NOT MATCHED BY TARGET significa que hay filas en la fuente que no están en la tabla destino.

NOT MATCHED BY SOURCE indica filas en la tabla destino que no están en la fuente.

La sentencia **MERGE** realiza las operaciones apropiadas para mantener la sincronización entre las dos tablas. Ten en cuenta que la condición **ON** en la cláusula **MERGE** especifica cómo se deben comparar las filas de las tablas fuente y destino para determinar si hay coincidencias.



Autoevaluación

Sesión 11



Pregunta 1

¿Cuál es el propósito principal de la declaración MERGE en SQL Server?

- ☐ Realizar consultas SELECT
- ☐ Combina las operaciones INSERT, UPDATE y DELETE en una sola declaración.
- ☐ Ejecutar procedimientos almacenados.
- ☐ Ninguna.

Pregunta 2

¿En qué escenario la declaración MERGE sería especialmente útil?

- ☐ Solo para consultas de selección de datos.
- ☐ Cuando se requiere combinar operaciones de inserción, actualización y eliminación basadas en una condición de coincidencia.
- ☐ En la ejecución de funciones definidas por el usuario.
- ☐ Todas.

Pregunta 3

¿Cuál es la función de la condición de coincidencia en la declaración MERGE?

- ☐ Determinar la tabla de origen.
- ☐ Especificar la tabla de destino.
- ☐ Definir la lógica para identificar registros que coinciden entre la tabla de origen y la de destino.
- ☐ Todas.

Pregunta 4

¿Qué tipo de operaciones se pueden realizar en la declaración MERGE?

- ☐ Solo operaciones de inserción.
- ☐ Solo operaciones de actualización.
- ☐ Operaciones de inserción, actualización y eliminación.
- ☐ Ninguna.

Autoevaluación
¡Vamos por más logros!

¡Felicitaciones!
Ha concluido la autoevaluación



Conclusiones

La declaración **MERGE** en SQL Server ofrece una eficiente combinación de operaciones **INSERT**, **UPDATE** y **DELETE** en una sola instrucción. Esto simplifica y optimiza el proceso de mantenimiento de datos en tablas relacionadas.

El uso de **MERGE** elimina la necesidad de realizar consultas separadas para inserciones, actualizaciones y eliminaciones, proporcionando un enfoque más limpio y conciso para gestionar cambios en los datos.

MERGE es especialmente útil en escenarios donde se necesita sincronizar datos entre tablas, por ejemplo, entre una tabla de origen y una tabla de destino, asegurando que ambas estén alineadas según una condición específica.

Al implementar adecuadamente la declaración **MERGE**, se pueden lograr mejoras significativas en el rendimiento. La optimización incluye el uso de índices, la estructura de la consulta y la consideración de la cantidad de registros afectados.

MERGE es eficaz en procesos de alimentación de datos, como la carga incremental de datos desde fuentes externas. Permite actualizar registros existentes, insertar nuevos registros y eliminar los obsoletos en una sola operación.



Aplicando lo aprendido:

Desarrollar la Guía de Laboratorio N°11

Referencias

CAPACHO, José y Wilson NIETO. Diseño de Bases de Datos [en línea]. Barranquilla: Universidad del Norte, 2017. ISBN 9789587418255. Disponible en: <https://search.ebscohost.com/login.aspx?direct=true&db=e000xww&AN=1690049&lang=es&site=ehost-live>

WANUMEN Luis, RIVAS Edwin, Mosquera Darín. Bases de datos en SQL Server [en línea]. Bogotá: Ecoe Ediciones, 2017. ISBN 9789587715705. Disponible en: <https://www.digitaliapublishing.com/a/66605>

HUESO Luis. Bases de datos [en línea]. Madrid: Rama Editorial, 2014. ISBN 9788499641577. Disponible en: <https://www.digitaliapublishing.com/a/109943>

PRIETO, Rafael. SGBD e instalación: administración de bases de datos (UF1469) [en línea]. Antequera : IC Editorial. ISBN 9788416433360. Disponible en: <https://www.digitaliapublishing.com/a/86830>





Pregrado