

## Tema 4

# Recuperación de Desastre



# Objetivos y resultados de aprendizaje

- El objetivo del presente tema es adquirir las habilidades necesarias para planear y ejecutar una estrategia adecuada de respaldo de datos y recuperación de desastres.
- Los objetivos específicos consisten en:
  - ✓ Conocer los tipos de Backup y Restores.
  - ✓ Establecer estrategias adecuadas para cada situación.
  - ✓ Manejo de utilidades relacionadas
    - Carga de ficheros
    - Importación / Exportación de datos

# Evaluación del tema

- Los resultados de aprendizaje correspondiente a este tema se evaluarán con los siguientes tipos de pruebas:
  - ✓ Prueba escrita de carácter teórico
  - ✓ Entrega de Práctica (final Tema 6)
  - ✓ Participación en clase

# Bibliografía

- Para obtener más información puedes consultar:
  - ✓ Libros en pantalla de SQL Server.
  - ✓ Bibliografía incluida en la guía didáctica
  - ✓ [https://www.guillesql.es/Articulos/Corregir Usuarios Huerfanos Instancia SQLServer.html](https://www.guillesql.es/Articulos/Corregir%20Usuarios%20Huerfanos%20Instancia%20SQLServer.html)

# Índice de contenidos

- ✓ Tipos Backup
- ✓ Modelo de Recuperación
- ✓ Estrategias de Backup
- ✓ Ejecución Backup y Restore

# Tipos de Backup

- Con frecuencia, los datos de una base de datos son la clave de las operaciones de una organización. Sin la base de datos, la organización no puede funcionar correctamente.
- Es importante que cada sistema de base de datos crítica para el funcionamiento de la organización disponga de los procedimientos de recuperación adecuados ante un desastre.
- Las operaciones de copia de seguridad y de restauración son una parte vital de la administración de datos y resultan esenciales para la recuperación de errores y desastres.
- Por consiguiente, gran parte de la función de un administrador de bases de datos consiste en garantizar que se efectúa una copia de seguridad de los datos para que éstos se puedan restaurar rápidamente en caso de ocurrir un desastre

# Tipos de Backup

- SQL Server proporciona varios métodos de copia de seguridad para satisfacer las necesidades de una amplia gama de entornos empresariales y actividades de base de datos

Tipo	Descripción
Completa	Todos los archivos de datos y parte del registro de transacciones
Registro de transacciones	Cualquier cambio en la base de datos registrado en los archivos de registro
Registro después del error	La porción activa del registro
Diferencial	Las partes de la base de datos que han cambiado desde la última copia de seguridad completa de la base de datos
Archivo / Grupo de archivos	Archivos o grupos de archivos especificados
Parcial	El grupo de archivos principal, todos los grupos de archivos de lectura/escritura y cualquier grupo de archivos especificado como de sólo lectura
Sólo de copia	La base de datos o registro (sin que afecte a la secuencia de copia de seguridad)

# Tipos de Backup

- Completa
  - ✓ Una copia de seguridad completa de una base de datos incluye los archivos de datos y parte del registro de transacciones.
  - ✓ Se usa como base en caso de producirse un error del sistema.
  - ✓ Al efectuar una copia de seguridad completa de la base de datos, SQL Server hace lo siguiente:
    - Realiza una copia de seguridad de cualquier actividad que haya tenido lugar durante la copia de seguridad.
    - Realiza una copia de seguridad de cualquier transacción no confirmada en el registro de transacciones.
  - ✓ SQL Server usa las partes del registro de transacciones que se capturaron en el archivo de copia de seguridad para garantizar la coherencia de los datos cuando se restaura la copia de seguridad.
  - ✓ La base de datos restaurada coincide con el estado de la base de datos cuando finalizó la copia de seguridad, a excepción de las transacciones no confirmadas. Cuando se recupera la base de datos, se deshacen las transacciones no confirmadas.



# Tipos de Backup

- Del Registro de Transacciones
  - ✓ Se registra cualquier cambio de la base de datos. Es Habitual realizarla junto con las copias de seguridad completas de la base de datos.
    - No se debe realizar copia del registro de transacciones a menos que haya realizado por lo menos una copia de seguridad completa de la base de datos.
    - No se pueden restaurar registros de transacciones sin la correspondiente copia de seguridad de la base de datos.
    - No se pueden realizar copias de seguridad de los registros de transacciones usando el modelo de recuperación simple.
  - ✓ Al realizarla SQL Server hace lo siguiente:
    - Realiza una copia de seguridad del registro de transacciones desde la última instrucción BACKUP LOG correctamente ejecutada hasta el final del registro de transacciones actual.
    - Trunca el registro de transacciones hasta el principio de la parte activa del registro de transacciones y descarta la información de la parte inactiva. de transacciones.
  - ✓ La parte activa del registro de transacciones empieza con la transacción abierta más antigua y continúa hasta el final del registro de transacciones.

# Tipos de Backup

- Del Registro de Transacciones Después de Error
  - ✓ Una copia de seguridad de registros después del error es una copia de seguridad del registro de transacciones que incluye la parte del registro del que previamente no se ha efectuado ninguna copia de seguridad (conocida como la parte activa del registro).
  - ✓ Este tipo de copia no trunca el registro y generalmente se usa cuando los archivos de datos para una base de datos se han convertido en inaccesibles pero el archivo de registro no está dañado.
- Diferencial
  - ✓ Debe realizar una copia de seguridad diferencial para minimizar el tiempo necesario para restaurar una base de datos modificada con frecuencia. Sólo puede realizar una copia de seguridad diferencial si ha realizado una copia de seguridad completa de la base de datos.
  - ✓ En una copia de seguridad diferencial, SQL Server hace lo siguiente:
    - Realiza una copia de seguridad de las partes de la base de datos que han cambiado desde la última copia de seguridad completa de la base de datos.
    - Realiza una copia de seguridad de cualquier actividad que tuvo lugar durante la copia de seguridad diferencial, así como cualquier transacción no confirmada en el registro de transacciones.

# Modelo de Recuperación

- SQL Server tiene tres modelos de recuperación de base de datos: simple, completa y de registro masivo. Cada modelo conserva los datos en caso de un error del servidor, pero presentan diferencias clave en el modo en que SQL Server recupera los datos.
- Puede establecer o cambiar en cualquier momento su modelo de recuperación, pero debe planear un modelo de recuperación cuando crea una base de datos.

Modelo de recuperación	Descripción
Simple	Usa copias completas o diferenciales de la base de datos. Trunca los registros de transacciones.
Completa	Incluye copias de seguridad de la base de datos y del registro de transacciones.
Registro masivo	Incluye copias de seguridad de la base de datos y del registro de transacciones, pero usa menos espacio de registro para ciertas operaciones.

# Modelo de Recuperación

- Modelo de Recuperación Simple
  - ✓ Normalmente se usa el modelo de recuperación simple para bases de datos pequeñas o en las que las modificaciones de datos no son frecuentes. Este modelo usa copias completas o diferenciales de la base de datos, y la recuperación se limita a la restauración de la base de datos hasta el momento en el que se efectuó la última copia de seguridad.
  - ✓ Todos los cambios realizados después de la copia de seguridad se perderán y deberán aplicarse de nuevo. El beneficio principal de este modelo es que ocupa menos espacio de almacenamiento para los registros y es el modelo más sencillo de implementar.

# Modelo de Recuperación

- Modelo de Recuperación Completo
  - ✓ Puede usar el modelo de recuperación completa cuando la recuperación completa de los medios dañados es la prioridad máxima. Este modelo usa copias de la base de datos y toda la información del registro para restaurar la base de datos. SQL Server registra todos los cambios de la base de datos, incluso las operaciones masivas y las creaciones de índices. En caso de que los registros en sí no estén dañados, SQL Server puede recuperar todos los datos excepto las transacciones que realmente estaban en proceso en el momento del error.
  - ✓ Dado que se registran todas las transacciones, la recuperación se puede llevar a cabo en cualquier momento. SQL Server admite la inserción de marcas con nombre en el registro de transacciones para permitir la recuperación hasta esa marca concreta.
  - ✓ Dado que las marcas de registros de transacción ocupan espacio en el registro, sólo deben usarse para transacciones que tengan una función importante en la estrategia de recuperación de la base de datos. La limitación principal de este modelo es el gran tamaño de los archivos de registro y los costos de almacenamiento y de rendimiento resultantes.

# Modelo de Recuperación

- Modelo de Recuperación de Registro Masivo
  - ✓ Similar al modelo de recuperación completo, la recuperación de registro masivo usa tanto las copias de seguridad de bases de datos como las copias de seguridad del registro para volver a crear una base de datos. Sin embargo, el modelo de recuperación de registro masivo usa menos espacio del registro para las operaciones siguientes:
    - CREATE INDEX / Operaciones de carga masiva / SELECT INTO / WRITETEXT / UPDATETEXT.
  - ✓ Estas operaciones sólo afectan al registro como bits en magnitudes en lugar de almacenar detalles de las operaciones en el registro.
  - ✓ Para conservar los cambios para una operación de carga masiva completa, las magnitudes que se marcan como modificadas también se almacenan en el registro. Como resultado de almacenar sólo el resultado final de varias operaciones, el registro suele ser más reducido y las operaciones masivas suelen ejecutarse más rápidamente.
  - ✓ Mediante el uso de este modelo pueden restaurarse todos los datos, pero una desventaja es que no es posible la restauración de sólo una parte de una copia de seguridad, como la restauración hasta una marca concreta.

# Estrategias de Backup

- Estrategia de copia de seguridad completa
  - ✓ Es un método de recuperación que implica la realización de copias de seguridad completas periódicas de la base de datos.
  - ✓ Si se produce un error en la base de datos, puede restaurar la copia de seguridad completa más reciente para recuperar el estado de la base de datos en el momento de la realización de esa copia de seguridad.
  - ✓ El tamaño de su base de datos y la frecuencia con la que se modifican los datos determinan el tiempo y los recursos implicados en la implementación de la estrategia de copia de seguridad completa de una base de datos.
  - ✓ Si se implementa y la base de datos está configurada para usar el modelo de recuperación completo o de recuperación por medio de registros de operaciones masivas, el fichero ldf acabará llenándose. Cuando el registro de transacciones está lleno, SQL Server puede impedir la actividad de la base de datos hasta que se vacíe este. Para evitar este problema, puede:
    - Establecer el modelo de recuperación de la base de datos como simple.
    - Borrar periódicamente el registro de transacciones mediante la opción NO\_LOG o TRUNCATE\_ONLY con la instrucción BACKUP LOG

# Estrategias de Backup

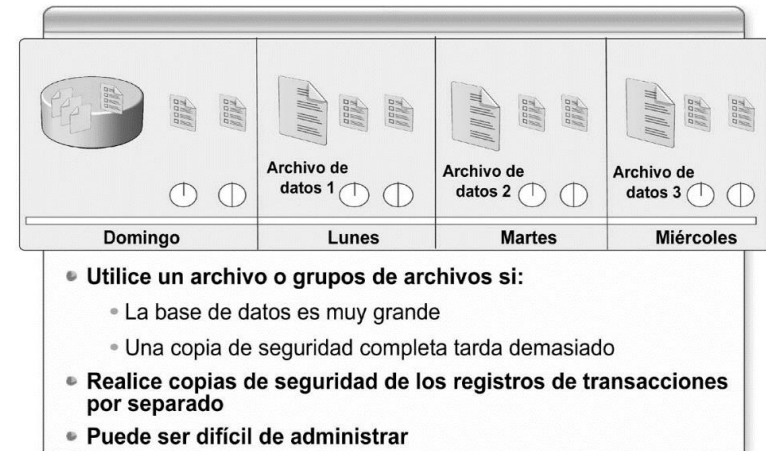
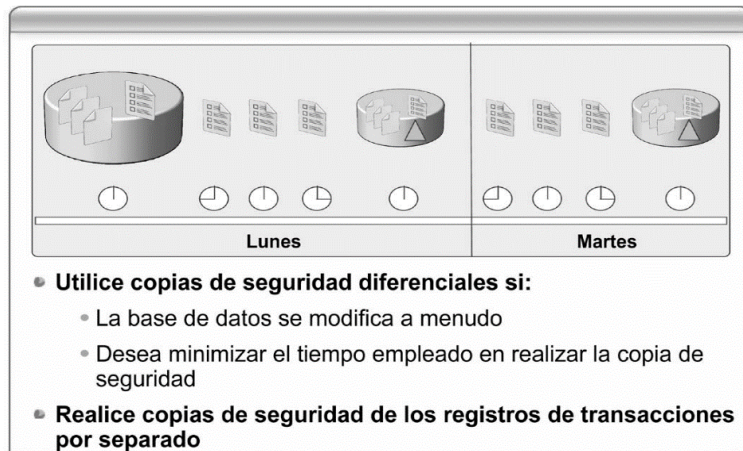
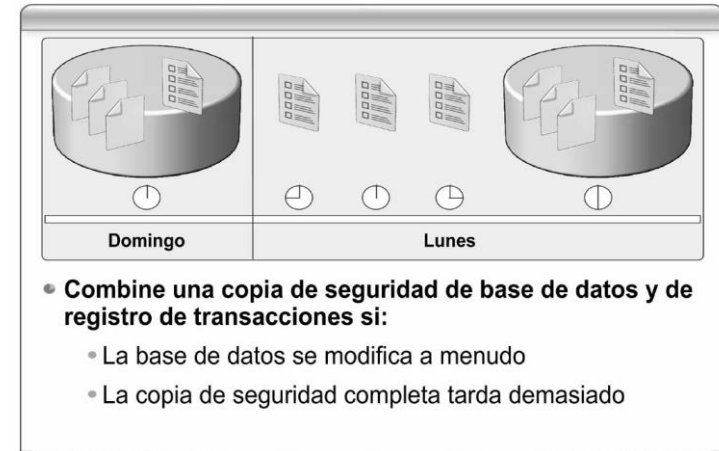
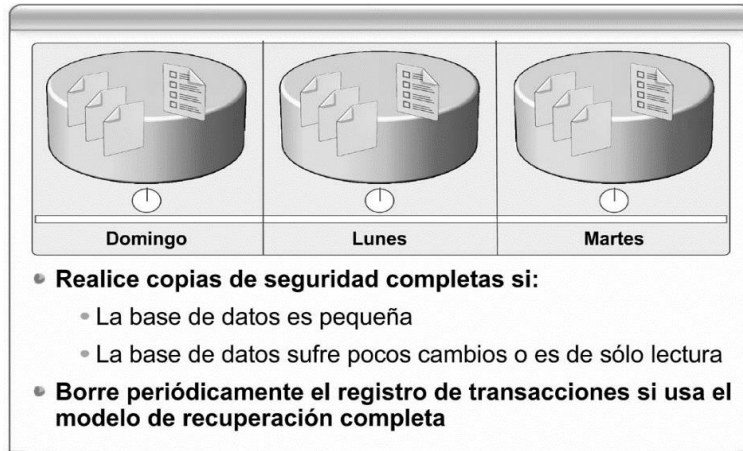
- Estrategia de copia de completa y del Registro de Transacciones
  - ✓ Cuando no resulte práctico cumplir sus requisitos de recuperación sólo mediante las copias de seguridad completas de la base de datos, se deben realizar copias de seguridad del registro de transacciones para obtener un registro de todas las actividades de base de datos que tuvieron lugar entre las copias de seguridad completas. Este enfoque se conoce como una estrategia de copia de seguridad de base de datos y de registro de transacciones.
  - ✓ Cuando se implementa una estrategia de copia de seguridad de base de datos y de registro de transacciones, puede restaurarse una base de datos desde la copia de seguridad completa de la base de datos más reciente y, a continuación, aplicar todas las copias de seguridad del registro de transacciones que se crearon desde la última copia de seguridad completa de la base de datos.
  - ✓ Implemente una estrategia de copia de seguridad completa de base de datos y de registro de transacciones para las bases de datos modificadas con frecuencia. También debe considerar si puede efectuarse la copia de seguridad de la base de datos y de los registros de transacciones en un período de tiempo razonable.



# Estrategias de Backup

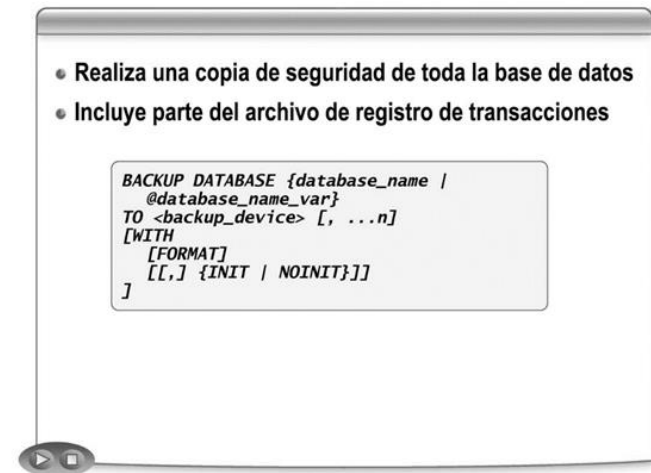
- Estrategia de copia de seguridad diferencial
  - ✓ Esta estrategia implica la realización de copias de seguridad completas periódicas de la base de datos con la intervención de copias de seguridad diferenciales. Opcionalmente, también puede realizar copias de seguridad de los registros de transacciones entre las copias de seguridad completa y diferencial.
  - ✓ Para recuperar la base de datos en caso de desastre, debe restaurar la copia de seguridad completa más reciente de la base de datos, seguida por la copia de seguridad diferencial más reciente, y, a continuación, restaurar cada registro de transacciones desde la última copia de seguridad diferencial realizada correctamente.
  - ✓ Use esta estrategia para reducir el tiempo de recuperación si se daña la base de datos. Por ejemplo, en lugar de aplicar varios registros de transacciones de gran tamaño, puede usarse una copia de seguridad diferencial para aplicar los cambios que se realizaron en la base de datos desde la última copia de seguridad completa de la base de datos.

# Estrategias de Backup (Resumen)



# Ejecución Backup y Restore

- ✓ Se puede realizar una copia de seguridad de la base de datos haciendo clic con el botón secundario en una base de datos en el Explorador de objetos, señalando Tareas y haciendo clic en Copia de seguridad.
- ✓ En el cuadro de diálogo Copia de seguridad de base de datos resultante, podemos seleccionar las opciones requeridas según las necesidades.
- ✓ Alternativamente, puede realizar una copia de seguridad completa de la base de datos mediante el uso de la instrucción BACKUP DATABASE.
- ✓ Una de las posibilidades si debe sobrescribirse o anexarse a un archivo de copia de seguridad. La opción predeterminada de SQL Server anexa las copias de seguridad (NOINIT) a un archivo.
- ✓ Alternativamente, puede realizar una copia de seguridad completa de la base de datos mediante el uso de la instrucción BACKUP DATABASE.
- ✓ Una de las posibilidades si debe sobrescribirse o anexarse a un archivo de copia de seguridad. La opción predeterminada de SQL Server anexa las copias de seguridad (NOINIT) a un archivo.



# Ejecución Backup y Restore

- Ejemplo\_1: Crea un dispositivo de copia de seguridad con el nombre lógico AdventureWorksBack y realiza una copia de seguridad completa de la base de datos.

*USE master*

*EXEC sp\_addumpdevice 'disk', 'AdventureWorksBack',  
'D:\MyBackupDir\AdventureWorksBack.bak'*

*BACKUP DATABASE AdventureWorks TO AdventureWorksBack*

- Ejemplo\_2: Realiza una copia de seguridad completa de la base de datos en el archivo AdventureWorksBack y sobrescribe cualquier copia de seguridad anterior de ese archivo

*BACKUP DATABASE AdventureWorks TO AdventureWorksBack WITH INIT*

- Ejemplo\_3: anexa una copia de seguridad completa de la base de datos al archivo AdventureWorksBack. Todos los archivos de copia de seguridad anteriores se mantienen intactos.

*BACKUP DATABASE AdventureWorks TO AdventureWorksBack WITH NOINIT*

# Ejecución Backup y Restore

- Ejemplo\_4: Crea un archivo de disco de copia de seguridad y realiza una copia de seguridad completa de la base de datos a ese archivo.

*BACKUP DATABASE AdventureWorks TO  
DISK = 'D:\Temp\MyTempBackup.bak'*

- Ejemplo\_5: Crea una copia de seguridad diferencial en un archivo de copia de seguridad temporal:

*BACKUP DATABASE AdventureWorks TO  
DISK = 'D:\MyData\MyDiffBackup.bak'  
WITH DIFFERENTIAL*

- Ejemplo\_6: una copia de seguridad de la base de datos AdventureWorks en un dispositivo denominado AWBackup y realizar suma de comprobación antes de escribir en los medios:

*BACKUP DATABASE [AdventureWorks] TO AWBackup WITH CHECKSUM*

# Ejecución Backup y Restore

- Ejemplo\_7: Comprueba la copia de seguridad completa de la base de datos AdventureWorks, verificando la legibilidad de todos los volumens, la suma de comprobación (si es que existe) y asegurando que existe espacio en el destino:

*RESTORE VERIFYONLY FROM AWBackup*

- Ejemplo\_8: Crea un dispositivo de copia de seguridad para el registro y realiza una copia de seguridad del registro de transacciones de la base de datos AdventureWorks.

*USE master*

*EXEC sp\_addumpdevice 'disk', 'AWBackLog', 'D:\Backup\AWBackLog.bak'*

*BACKUP LOG AdventureWorks TO AWBackLog*

- Ejemplo\_9: Copia de seguridad del registro de transacciones, después del error:

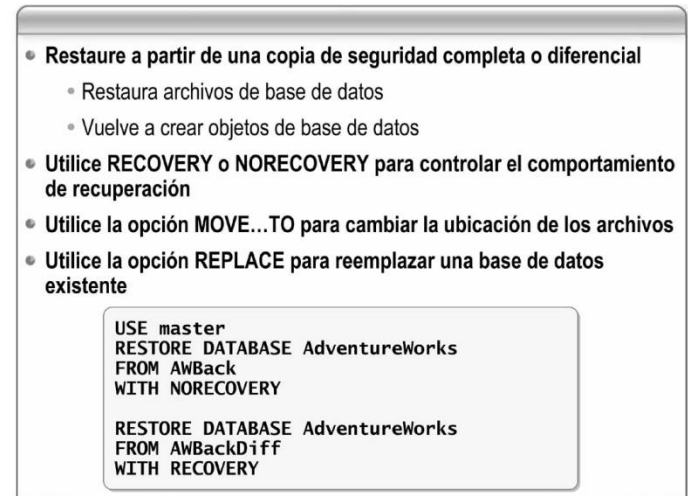
*BACKUP LOG AdventureWorks*

*TO 'C:\Backup\AWTail.bak'*

*WITH NORECOVERY, NO\_TRUNCATE*

# Ejecución Backup y Restore

- ✓ Cuando se restaura una base de datos desde una copia de seguridad de la base de datos, SQL Server recrea la base de datos y todos sus archivos asociados y, a continuación, los coloca en su ubicación original. Todos los objetos de base de datos se vuelven a crear automáticamente. No es necesario volver a generar el esquema de la base de datos antes de restaurar la base de datos.
- ✓ Si implementa una estrategia de copia de seguridad completa de la base de datos y no tiene ningún registro de transacciones o copia de seguridad diferencial, especifique la opción RECOVERY.
- ✓ Si existe alguna copia de seguridad del registro de transacciones o copia de seguridad diferencial, especifique la opción NORECOVERY para posponer el proceso de recuperación hasta que se restaure la última copia de seguridad.



# Ejecución Backup y Restore

- Ejemplo\_10: Se da por supuesto que existe una copia de seguridad completa en el archivo de copia de seguridad permanente AWBack. El archivo AWBackDiff contiene una copia de seguridad diferencial.

*USE master*

*RESTORE DATABASE AdventureWorks*

*FROM AWBack WITH NORECOVERY*

*RESTORE DATABASE AdventureWorks*

*FROM AWBackDiff*

*WITH RECOVERY*



# Ejecución Backup y Restore

- ✓ Normalmente, los registros de transacciones se restauran para aplicar los cambios en la base de datos desde la última copia de seguridad completa o diferencial. Además, pueden restaurarse registros de transacciones para recuperar una base de datos hasta un momento dado.
- ✓ Para garantizar la coherencia de los datos puede que tenga que restaurar las copias de seguridad de registros de transacciones adicionales creadas tras una copia de seguridad diferencial.
- ✓ Debe restaurar primero toda la copia de seguridad completa de la base de datos, seguida de la copia de seguridad diferencial más reciente, en caso de que exista. A continuación, debe restaurar en el mismo orden todas las copias de seguridad de registros de transacciones desde la última copia de seguridad completa o diferencial. Cuando hay varios registros de transacciones que deben aplicarse, especifique la opción **NORECOVERY** para todos los registros de transacciones exceptuando el último.

- **Restaura a partir de una copia de seguridad del registro de transacciones**
  - Restaura los cambios en la base de datos que están registrados en el registro de transacciones
- **Utilice la opción **RECOVERY** en el registro final que se va a restaurar**

```
RESTORE DATABASE AdventureWorks FROM AWBack  
WITH NORECOVERY
```

```
RESTORE LOG AdventureWorks FROM AWBackLog  
WITH FILE = 1, NORECOVERY
```

```
RESTORE LOG AdventureWorks FROM AWBackLog  
WITH FILE = 2, RECOVERY
```

# Ejecución Backup y Restore

- Ejemplo\_11: Se da por supuesto que existe una copia de seguridad y que existen dos copias de seguridad completa y otra del registro de transacciones en otro archivo. Se deben llevar a cabo tres operaciones de restauración independientes para garantizar la coherencia de la base de datos

- ✓ A.- Restaure la base de datos desde una copia de seguridad completa

*USE master*

*RESTORE DATABASE AdventureWorks  
FROM AWBack WITH NORECOVERY*

- ✓ B.- Restaure el primer registro de transacciones

*USE master*

*RESTORE LOG AdventureWorks  
FROM AWBackLog  
WITH FILE = 1, NORECOVERY*

- ✓ C.- Restaure el segundo registro de transacciones, La opción RECOVERY devuelve la base de datos AdventureWorks a un estado coherente (Se ponen al día las transacciones confirmadas y, a continuación, deshaga todas las transacciones no confirmadas)

*USE master*

*RESTORE LOG AdventureWorks  
FROM AWBackLog  
WITH FILE = 2, RECOVERY*