
Manual Técnico Avanzado: Alta Disponibilidad y Recuperación ante Desastres (HA/DR)

Semana 16 | SQL Server 2019/2022

1. Prerrequisitos del Entorno

Para ejecutar este laboratorio de Alta Disponibilidad (HA) y Recuperación ante Desastres (DR), se requiere la siguiente infraestructura mínima.

Infraestructura Necesaria

- Instancias de SQL Server:** Dos instancias instaladas (pueden estar en la misma máquina física para pruebas, o en dos VMs separadas).
 - Principal (Source): **localhost** (o nombre de tu servidor).
 - Secundaria (Destination): **localhost\instancia2** (o nombre de la segunda instancia).
- Carpeta Compartida (Network Share):** Una carpeta en Windows accesible por ambas instancias con permisos de **Lectura/Escritura** para el usuario de servicio de SQL Server.
 - Ejemplo: **C:\LS_Backups** (compartida como **\\TuPC\LS_Backups**).
- Edición de SQL Server:** Developer, Enterprise o Standard (Standard tiene limitaciones en AlwaysOn, pero soporta Log Shipping).

Script de Base de Datos: **EmpresaDemo**

Ejecuta el siguiente script en tu instancia **Principal** para preparar el entorno.

NOTA IMPORTANTE: El modelo de recuperación debe ser **FULL** para que funcionen las estrategias de Log Shipping o AlwaysOn.

SQL

USE master;

GO

-- 1. Crear la Base de Datos

IF NOT EXISTS (SELECT name FROM sys.databases WHERE name = N'EmpresaDemo')
BEGIN

CREATE DATABASE [EmpresaDemo];

END

GO

USE [EmpresaDemo];

GO

```

-- 2. Configurar Modelo de Recuperación a FULL (Crítico para HA/DR)
ALTER DATABASE [EmpresaDemo] SET RECOVERY FULL;
GO

-- 3. Crear Tablas de Ejemplo
CREATE TABLE Departamentos (
    DeptID INT IDENTITY(1,1) PRIMARY KEY,
    Nombre VARCHAR(50) NOT NULL,
    Presupuesto DECIMAL(10,2)
);

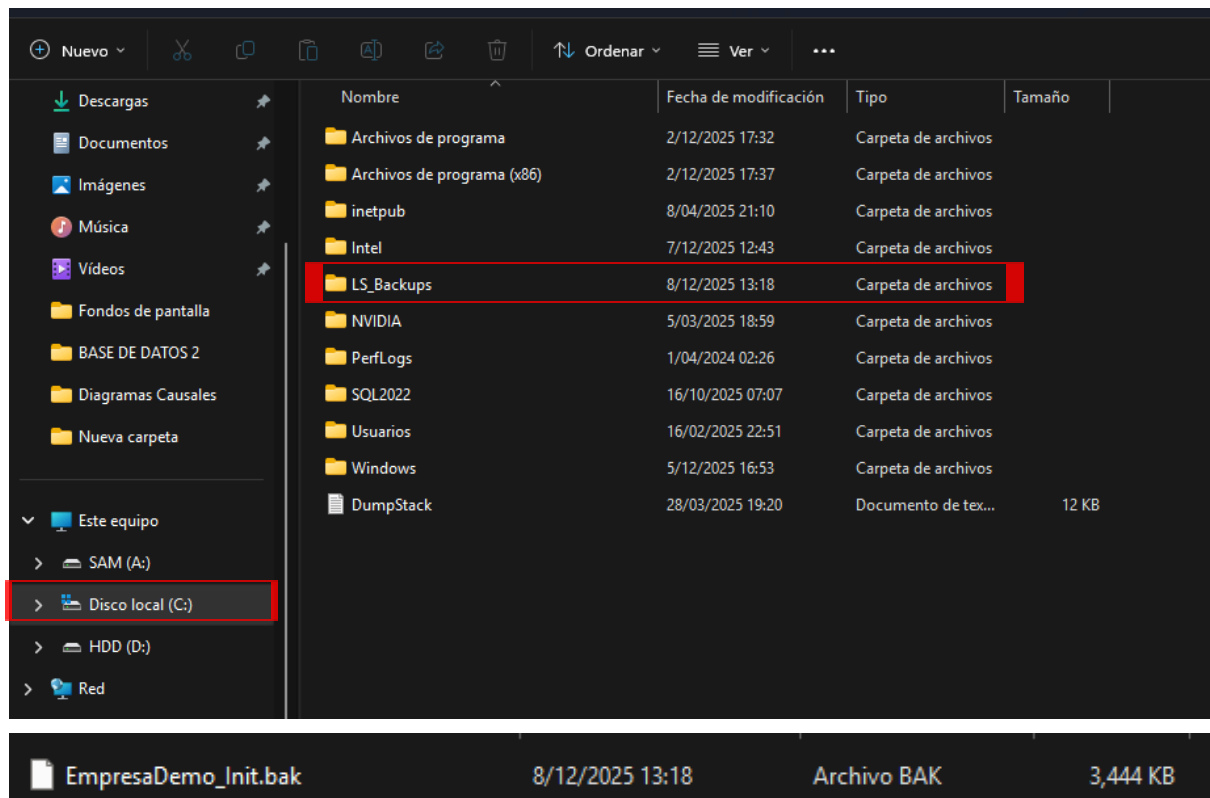
CREATE TABLE Empleados (
    EmpID INT IDENTITY(1,1) PRIMARY KEY,
    Nombre VARCHAR(100) NOT NULL,
    Salario DECIMAL(10,2),
    DeptID INT FOREIGN KEY REFERENCES Departamentos(DeptID),
    FechaIngreso DATETIME DEFAULT GETDATE()
);

-- 4. Insertar Datos Semilla
INSERT INTO Departamentos (Nombre, Presupuesto) VALUES
('IT', 500000.00), ('Recursos Humanos', 150000.00), ('Ventas', 300000.00);

INSERT INTO Empleados (Nombre, Salario, DeptID) VALUES
('Ana Lopez', 4500.00, 1),
('Carlos Ruiz', 2800.00, 2),
('Maria Gomez', 3200.00, 3);

-- 5. Realizar el primer Full Backup (Obligatorio para iniciar la cadena de logs)
-- Asegúrate de que la ruta exista en tu disco C:
BACKUP DATABASE [EmpresaDemo]
TO DISK = 'C:\LS_Backups\EmpresaDemo_Init.bak'
WITH INIT;
GO

```



2. Fundamentos Teóricos: Estrategias de HA/DR

Definiciones Clave

- Log Shipping (Envío de Registros):**
 - Mecanismo donde el servidor principal realiza backups del log de transacciones periódicamente, los copia al servidor secundario y los restaura. Existe un desfase (latencia) entre ambos.
- Replication (Replicación):**
 - Mueve datos y objetos (tablas, vistas) de una base a otra. Puede ser a nivel de objeto, no necesariamente toda la base de datos. Útil para distribuir datos a sucursales.
- AlwaysOn Availability Groups (Grupos de Disponibilidad):**
 - La solución moderna y robusta. Permite tener un grupo de bases de datos que conmutan por error juntas. Requiere un **Clúster de Conmutación por Error de Windows (WSFC)** y Active Directory. Ofrece réplicas de lectura en tiempo real.

Tabla Comparativa

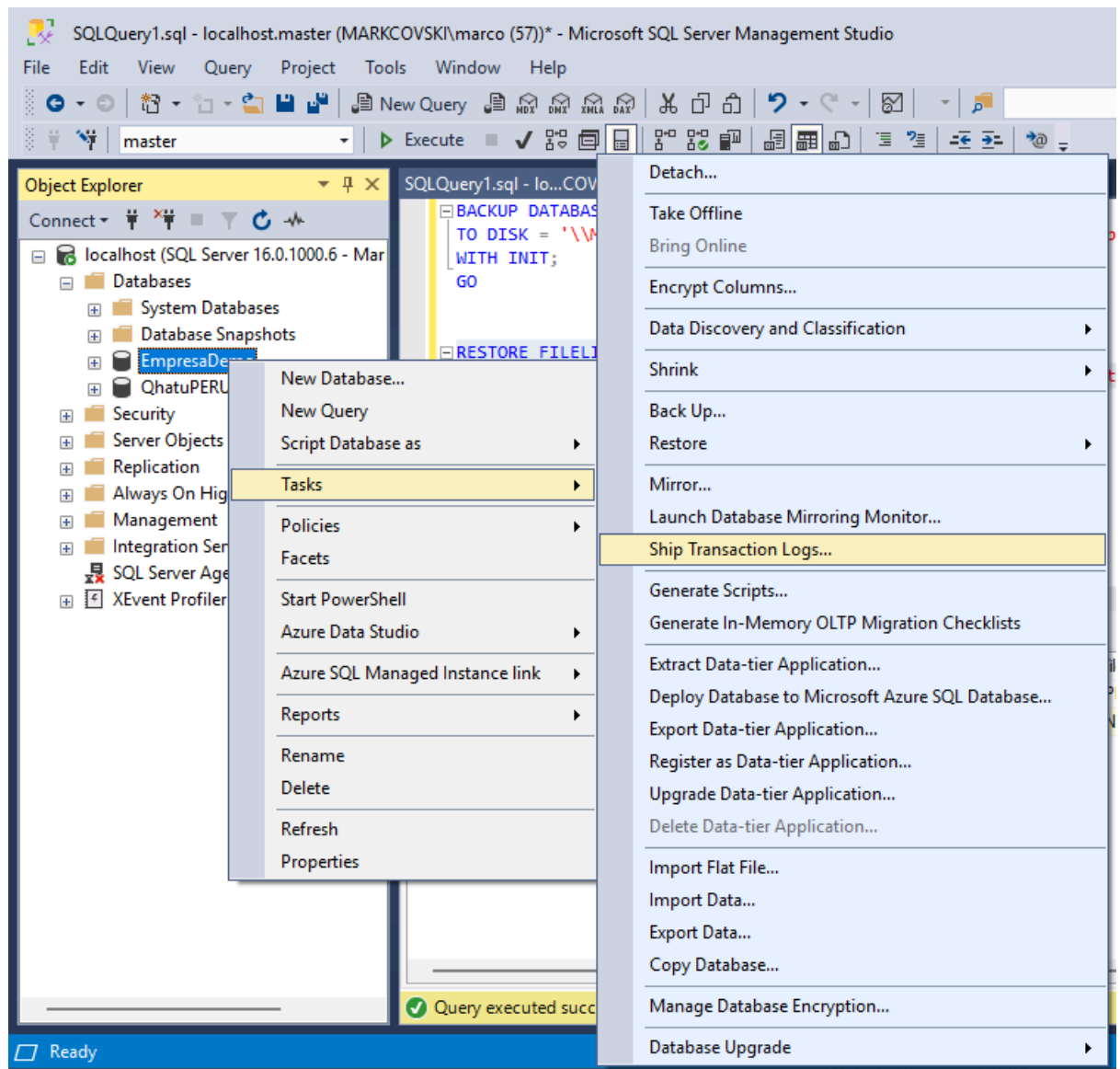
Característica	Log Shipping	Replicación	AlwaysOn AG
Nivel de Protección	Base de Datos completa.	Objetos específicos (Tablas).	Grupo de Bases de Datos.
Failover (Conmutación)	Manual.	No nativo para DR.	Automático (con WSFC).
Latencia de Datos	Alta (depende de la frecuencia del backup).	Baja (casi tiempo real).	Síncrona (Cero pérdida) o Asíncrona.
Requisitos Infra.	Bajo (Solo carpeta compartida).	Medio.	Alto (Requiere Dominio + Cluster Windows).
Uso Principal	DR (Recuperación ante desastres) barato.	Distribución de datos / Reportes.	Alta Disponibilidad Crítica (HA) y DR.

3. Guía de Implementación Práctica: Log Shipping

*Dado que AlwaysOn requiere un entorno de Dominio complejo, configuraremos **Log Shipping**, que es la base fundamental para entender cómo viajan los logs de transacción.*

Configuración en SSMS

1. Abre **SQL Server Management Studio (SSMS)** y conéctate a la instancia **Principal**.
2. En el **Explorador de Objetos**, haz clic derecho sobre la base de datos **EmpresaDemo**.
3. Selecciona **Tasks (Tareas) > Ship Transaction Logs... (Enviar registros de transacciones)**.
4. Marca la casilla: *"Enable this as a primary database in a log shipping configuration"*.



Configurar Backup y Secundario

1. **Backup Settings:** Haz clic y especifica la ruta de red donde se guardarán los backups (ej: `\\TuPC\LS_Backups`). Configura el job para que corra cada 5 minutos (para ver resultados rápidos en el laboratorio).

Transaction Log Backup Settings

Transaction log backups are performed by a SQL Server Agent job running on the primary server instance.

Network path to backup folder (example: \\filesrvr\backup):
//MARKCOVSKI/LogShipping_Principal

If the backup folder is located on the primary server, type a local path to the folder (example: c:\backup):
C:\LS_Backups

Note: you must grant read and write permission on this folder to the SQL Server service account of this primary server instance. You must also grant read permission to the proxy account for the copy job (usually the SQL Server Agent service account for the secondary server instance).

Delete files older than: 72 Hours

Alert if no backup occurs within: 1 Hours

Backup job

Job name: LSBackup_EmpresaDemo Schedule...

Schedule: Occurs every day every 5 minute(s) between 00:00:00 and 23:59:00. Schedule will be used starting on 8/12/2025. ☐ Disable this job

Compression

Set backup compression: Use the default server setting

Note: If you backup the transaction logs of this database with any other job or maintenance plan, Management Studio will not be able to restore the backups on the secondary server instances.

Help OK Cancel

2. **Secondary Databases:** Haz clic en **Add....**
3. Conéctate a tu **Instancia Secundaria.**

Secondary Database Settings

Secondary server instance: localhost Connect...

Secondary database: EmpresaDemo
Select an existing database or enter the name to create a new database.

Initialize Secondary Database Copy Files Restore Transaction Log

You must restore a full backup of the primary database into secondary database before it can be a log shipping destination.

Do you want the Management Studio to restore a backup into the secondary database?

☐ Yes, generate a full backup of the primary database and restore it into the secondary database (and create the secondary database if it doesn't exist) Restore Options...

☐ Yes, restore an existing backup of the primary database into the secondary database (and create the secondary database if it doesn't exist)

Specify a network path to the backup file that is accessible by the secondary server instance.

Backup file: Restore Options...


☒ No, the secondary database is initialized.

Help OK Cancel



4. En la pestaña **Initialize Secondary Database**, elige "Yes, generate a full backup...".
5. En la pestaña **Copy Files**, especifica una carpeta local en el servidor secundario para descargar los archivos.
6. En la pestaña **Restore Transaction Log**, selecciona **"No recovery mode"** o **"Standby mode"**.
 - *Nota:* Elige **Standby** si quieres poder leer datos en la secundaria mientras no se están restaurando logs.
7. Finaliza el asistente y asegúrate de que todos los pasos marquen "Success".

Save Log Shipping Configuration

Restoring backup to secondary database

 **Success** 2 Total 0 Error
2 Success 0 Warning

Details:

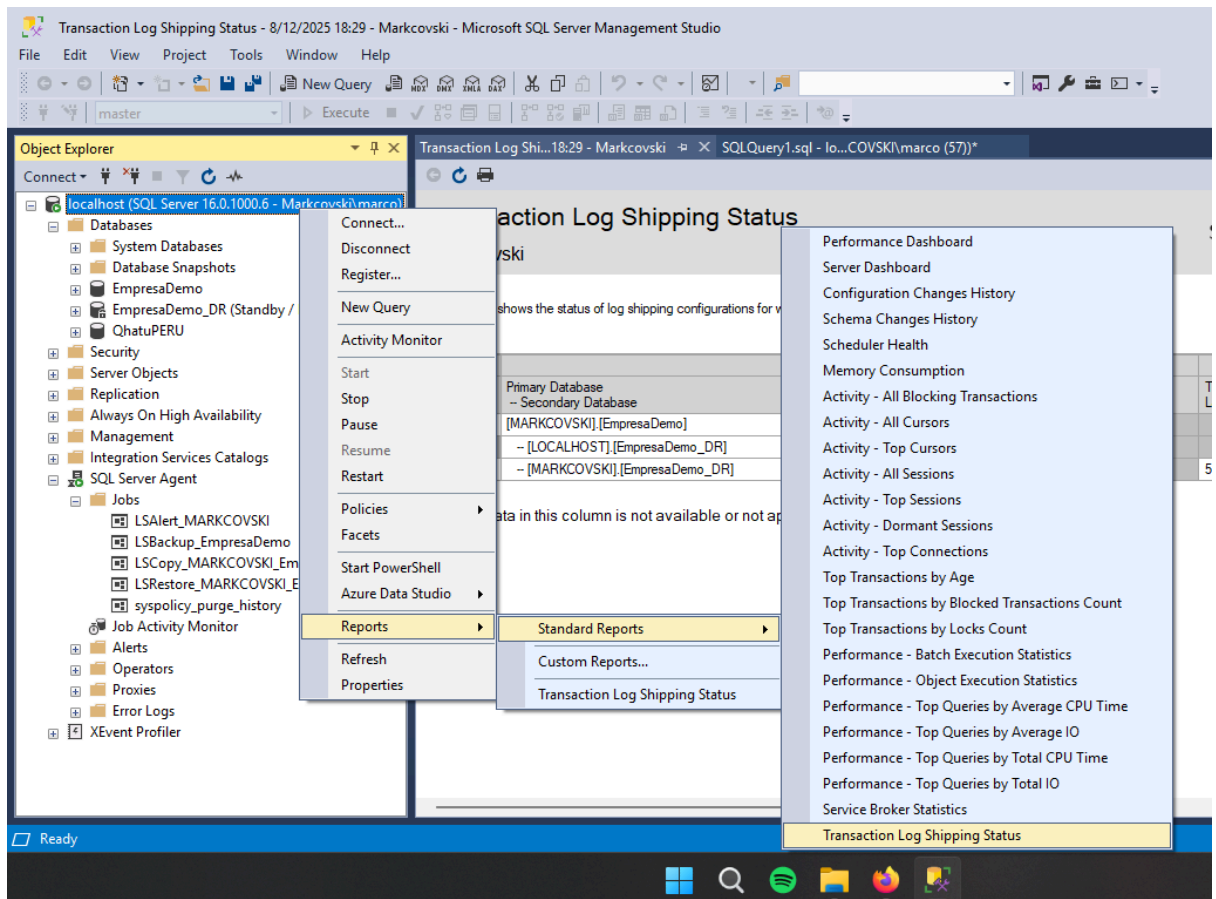
	Action	Status	Message
	Saving secondary destination configuration [loc...	Success	
	Saving primary backup setup	Success	

Close Report ▼

Verificación (SSMS v20 y T-SQL)

A. Verificación Visual en SSMS:

1. En la Instancia Principal, clic derecho en **EmpresaDemo** > **Reports** > **Standard Reports** > **Transaction Log Shipping Status**.
2. Debe mostrar el estado "Good" (Verde).



Transaction Log Shipping Status

SQL Server

Markcovski

This report shows the status of log shipping configurations for which this server instance is a primary, secondary, or monitor.

Status	Primary Database -- Secondary Database	Backup			Copy	Restore			
		Time Since Last	Threshold	Alert Enabled	Time Since Last	Time Since Last	Latency of Last File	Threshold	Alert Enabled
Good	[MARKCOVSKI].[EmpresaDemo]	5 min	60 min	True					
	-- [LOCALHOST].[EmpresaDemo_DR]								
Good	-- [MARKCOVSKI].[EmpresaDemo_DR]					5 min		45 min	True

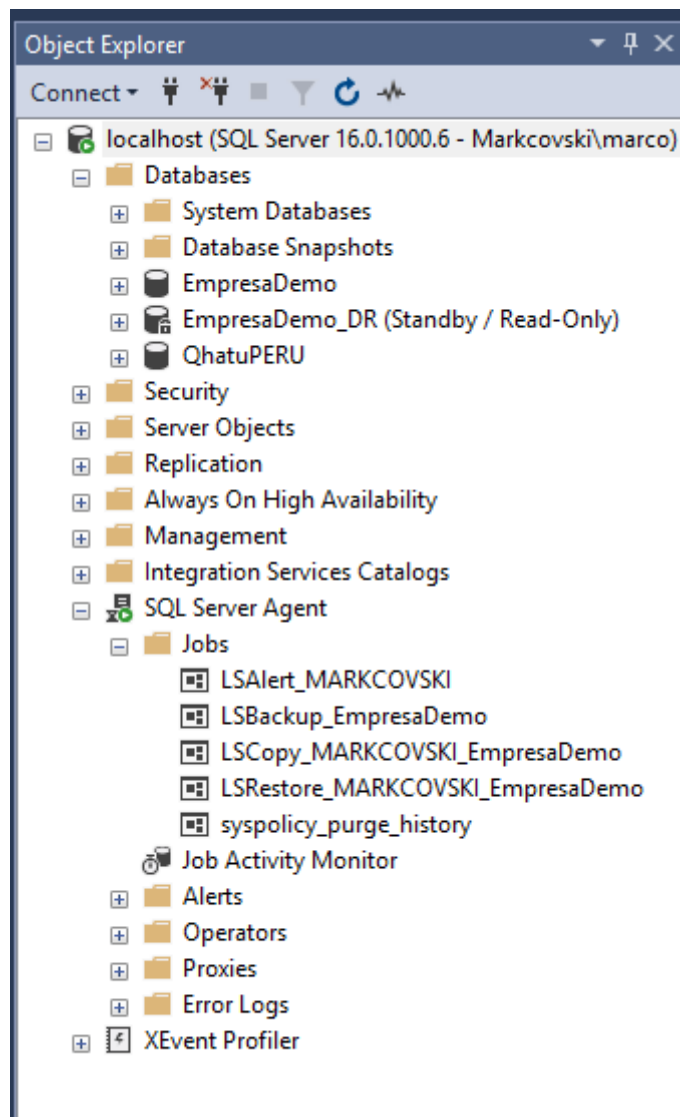
- data in this column is not available or not applicable for this server instance.

B. Verificación de Jobs (Agente SQL):

Ve al SQL Server Agent en el Explorador de Objetos. Deberás ver creados automáticamente:

- En Principal: **LSBackup_EmpresaDemo**

- En Secundaria: **LSCopy_...** y **LSRestore_...**



C. Verificación por Script:

Ejecuta esto en la Instancia Secundaria para ver si la base de datos está en modo de espera/restauración:

SQL

```
SELECT name, state_desc, is_in_standby
```

```
FROM sys.databases
```

```
WHERE name = 'EmpresaDemo';
```

-- Resultado esperado: state_desc = 'RESTORING' o 'ONLINE' (si es Standby)

The screenshot shows a SQL query window with the following text:

```
SELECT name, state_desc, is_in_standby
FROM sys.databases
WHERE name = 'EmpresaDemo';
```

Below the query window, the 'Results' tab is active, displaying a table with the following data:

	name	state_desc	is_in_standby
1	EmpresaDemo	ONLINE	0

4. Simulación de Failover (Laboratorio)

En este escenario, simularemos que el servidor Principal ha fallado catastróficamente y necesitamos activar el servidor Secundario.

Fase 1: Simulación del Fallo

1. Detén el servicio de SQL Server de la instancia Principal (o simplemente desconéctala en SSMS y no la uses).

USE master;

GO

-- 1. Simulamos que la base de datos principal se corrompió o desconectó

-- El "ROLLBACK IMMEDIATE" expulsa a cualquiera que esté conectado.

ALTER DATABASE [EmpresaDemo] SET OFFLINE WITH ROLLBACK IMMEDIATE;

GO

-- 2. PRUEBA DE QUE FALLÓ:

-- Intenta seleccionar algo. Debería darte error.

SELECT * FROM [EmpresaDemo].dbo.Empleados;

-- Resultado esperado: "Database 'EmpresaDemo' cannot be opened because it is offline."

The screenshot shows the 'Messages' window with the following text:

```
Se están revirtiendo las transacciones no calificadas. Estimación de porcentaje de reversión: 0%.
Se están revirtiendo las transacciones no calificadas. Estimación de porcentaje de reversión: 100%.
Msg 942, Level 14, State 4, Line 43
No se puede abrir la base de datos 'EmpresaDemo' por estar sin conexión.

Completion time: 2025-12-08T18:36:40.2653843-05:00
```

2. Intenta hacer un **INSERT** en la Principal; fallará.

Fase 2: Poner en línea la Secundaria (Failover Manual)

Dado que Log Shipping no es automático, debes ejecutar este script en la **Instancia Secundaria** para romper la cadena de logs y hacer la base operativa.

SQL

USE master;

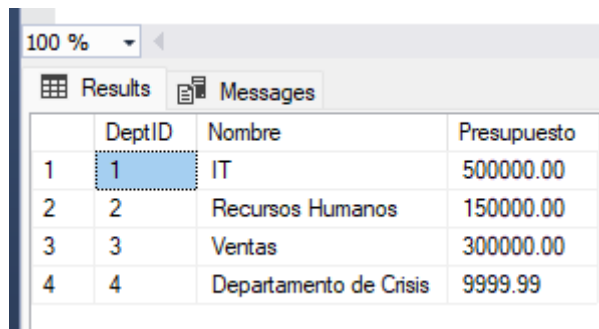
GO

```
-- 1. Promover la base de datos de respaldo a PRINCIPAL
-- Esto rompe la cadena de Log Shipping y la deja lista para escribir.
RESTORE DATABASE [EmpresaDemo_DR] WITH RECOVERY;
GO
```

```
-- 2. VALIDAR QUE AHORA ES LA NUEVA JEFA
-- Insertamos un dato en la base que antes era solo lectura.
USE [EmpresaDemo_DR];
GO
```

```
INSERT INTO Departamentos (Nombre, Presupuesto)
VALUES ('Departamento de Crisis', 9999.99);
```

```
SELECT * FROM Departamentos;
```



	DeptID	Nombre	Presupuesto
1	1	IT	500000.00
2	2	Recursos Humanos	150000.00
3	3	Ventas	300000.00
4	4	Departamento de Crisis	9999.99

Fase 3: Retorno a la Operación (Failback)

Nota Teórica: En Log Shipping, el "Failback" no es un botón. Una vez que la secundaria se vuelve primaria (escritura), la cadena se rompe. Para volver al estado original, debes:

1. Eliminar la configuración de Log Shipping antigua.
 2. Hacer un backup Full de la *nueva* primaria (la que era secundaria).
 3. Restaurarla en el servidor original y configurar Log Shipping en dirección inversa.
-