

UNIVERSIDAD PERUANA LOS ANDES

**FACULTAD INGENIERIA INGENIERÍA DE
SISTEMAS Y COMPUTACIÓN**



MANUAL

**RESPALDO Y RECUPERACION DE
BASE DE DATOS**

ASIGNATURA: BASE DE DATOS II

DOCENTE: FERNÁNDEZ BEJARANO RAUL

ESTUDIANTE: Bonifacio Hilario Erick

CÓDIGO: S01238F

HUANCAYO-2025

**Proyecto 1: Estrategia de backup completo diario y
verificación automática**

1. Enunciado

Implementar un backup completo diario de la base QhatuPeru, guardando el archivo con la fecha y verificando automáticamente con RESTORE VERIFYONLY.

2. Script T-SQL

```
DECLARE @fecha NVARCHAR(20) = CONVERT(NVARCHAR(20), GETDATE(), 112);  
DECLARE @ruta NVARCHAR(200) = 'C:\Backups\QhatuPeru_Full_' + @fecha  
+ '.bak';
```

```
BEGIN TRY
```

```
    BACKUP DATABASE QhatuPeru
```

```
    TO DISK = @ruta
```

```
    WITH INIT, FORMAT, CHECKSUM, STATS = 10;
```

```
    RESTORE VERIFYONLY
```

```
    FROM DISK = @ruta;
```

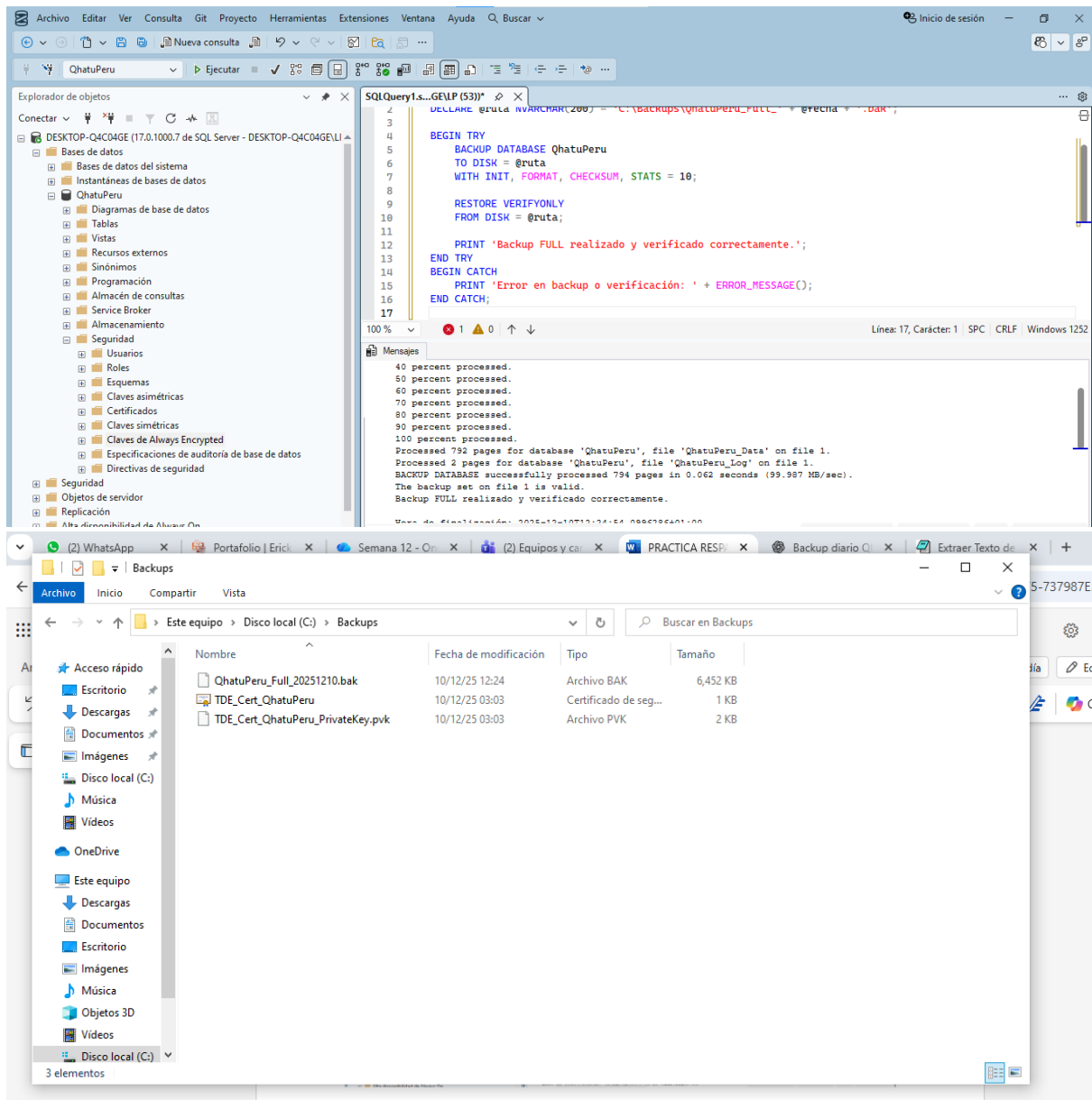
```
    PRINT 'Backup FULL realizado y verificado correctamente.';
```

```
END TRY
```

```
BEGIN CATCH
```

```
    PRINT 'Error en backup o verificación: ' + ERROR_MESSAGE();
```

```
END CATCH;
```



3. Justificación técnica

Se asegura una copia completa de la base y se verifica su integridad antes de considerarla válida. La fecha permite organización cronológica.

4. Buenas prácticas

- Nombres de archivo con fecha.
- Verificación automática (RESTORE VERIFYONLY).
- Control de errores (TRY...CATCH).
- Visualización de progreso (STATS = 10).

Proyecto 2: Backup full semanal + verificación automática

1. Enunciado

Backup completo semanal cada domingo a las 02:00, registrar resultado en tabla BackupAudit.

2. Script T-SQL

```
IF NOT EXISTS (SELECT * FROM sys.tables WHERE name = 'BackupAudit')
BEGIN
    CREATE TABLE BackupAudit (
        Id INT IDENTITY(1,1) PRIMARY KEY,
        TipoBackup VARCHAR(20),
        Fecha DATETIME,
        RutaBackup VARCHAR(200),
        Estado VARCHAR(50)
    );
END
GO

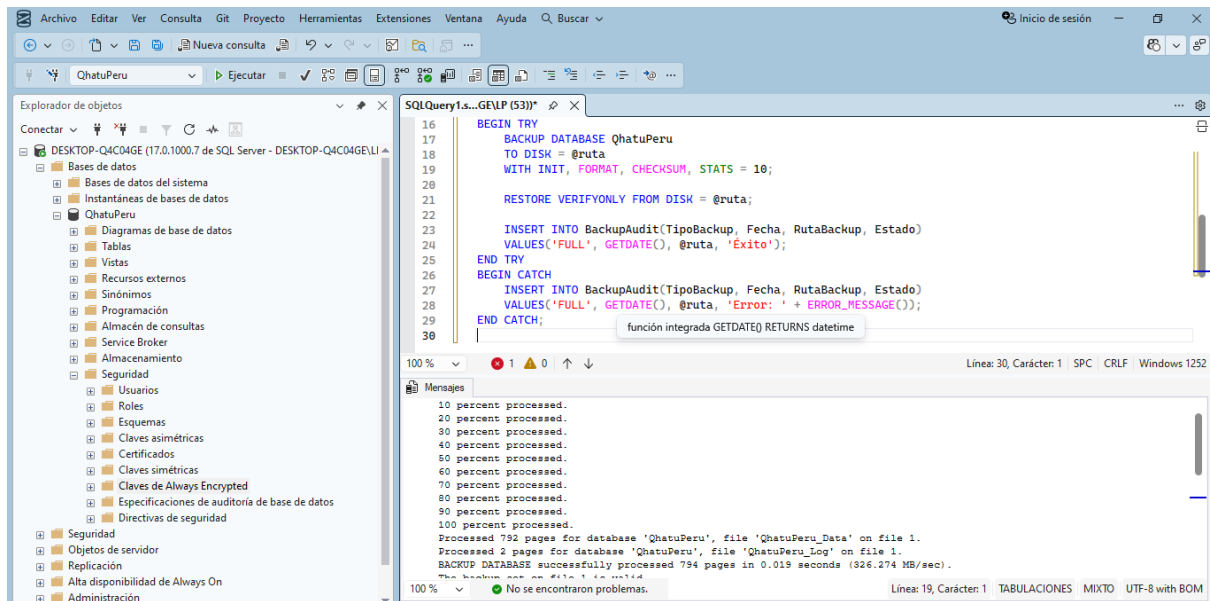
DECLARE @fecha NVARCHAR(20) = CONVERT(NVARCHAR(20), GETDATE(), 112);
DECLARE @ruta NVARCHAR(200) = 'C:\Backups\QhatuPeru_Full_' + @fecha
+ '.bak';

BEGIN TRY
    BACKUP DATABASE QhatuPeru
    TO DISK = @ruta
    WITH INIT, FORMAT, CHECKSUM, STATS = 10;

    RESTORE VERIFYONLY FROM DISK = @ruta;

    INSERT INTO BackupAudit(TipoBackup, Fecha, RutaBackup, Estado)
    VALUES('FULL', GETDATE(), @ruta, 'Éxito');
END TRY
BEGIN CATCH
    INSERT INTO BackupAudit(TipoBackup, Fecha, RutaBackup, Estado)
    VALUES('FULL', GETDATE(), @ruta, 'Error: ' + ERROR_MESSAGE());
```

END CATCH;



3. Justificación técnica

Se registra historial de backups y se garantiza que cada backup es verificable.

4. Buenas prácticas

- Auditoría de backups.
- Automatización y control de errores.
- Nombre de archivo con fecha.

Proyecto 3: Estrategia combinada: FULL semanal + DIF diario + LOG cada 30 minutos

1. Enunciado

Implementar FULL semanal, DIFERENCIAL diario y LOG cada 30 minutos, con posibilidad de restaurar hasta un punto en el tiempo.

2. Script T-SQL

```
DECLARE @fecha NVARCHAR(20) = CONVERT(NVARCHAR(20), GETDATE(), 112);
DECLARE @rutaFull NVARCHAR(200) = 'C:\Backups\QhatuPeru_Full_' +
@fecha + '.bak';
DECLARE @rutaDiff NVARCHAR(200) = 'C:\Backups\QhatuPeru_Diff_' +
@fecha + '.bak';
DECLARE @rutaLog NVARCHAR(200) = 'C:\Backups\QhatuPeru_Log_' +
```

```
@fecha + '.trn';
```

```
IF DATEPART(WEEKDAY, GETDATE()) = 1
```

```
BEGIN
```

```
    BACKUP DATABASE QhatuPeru
```

```
    TO DISK = @rutaFull
```

```
    WITH INIT, FORMAT, CHECKSUM, STATS = 10;
```

```
END
```

```
ELSE
```

```
BEGIN
```

```
    BACKUP DATABASE QhatuPeru
```

```
    TO DISK = @rutaDiff
```

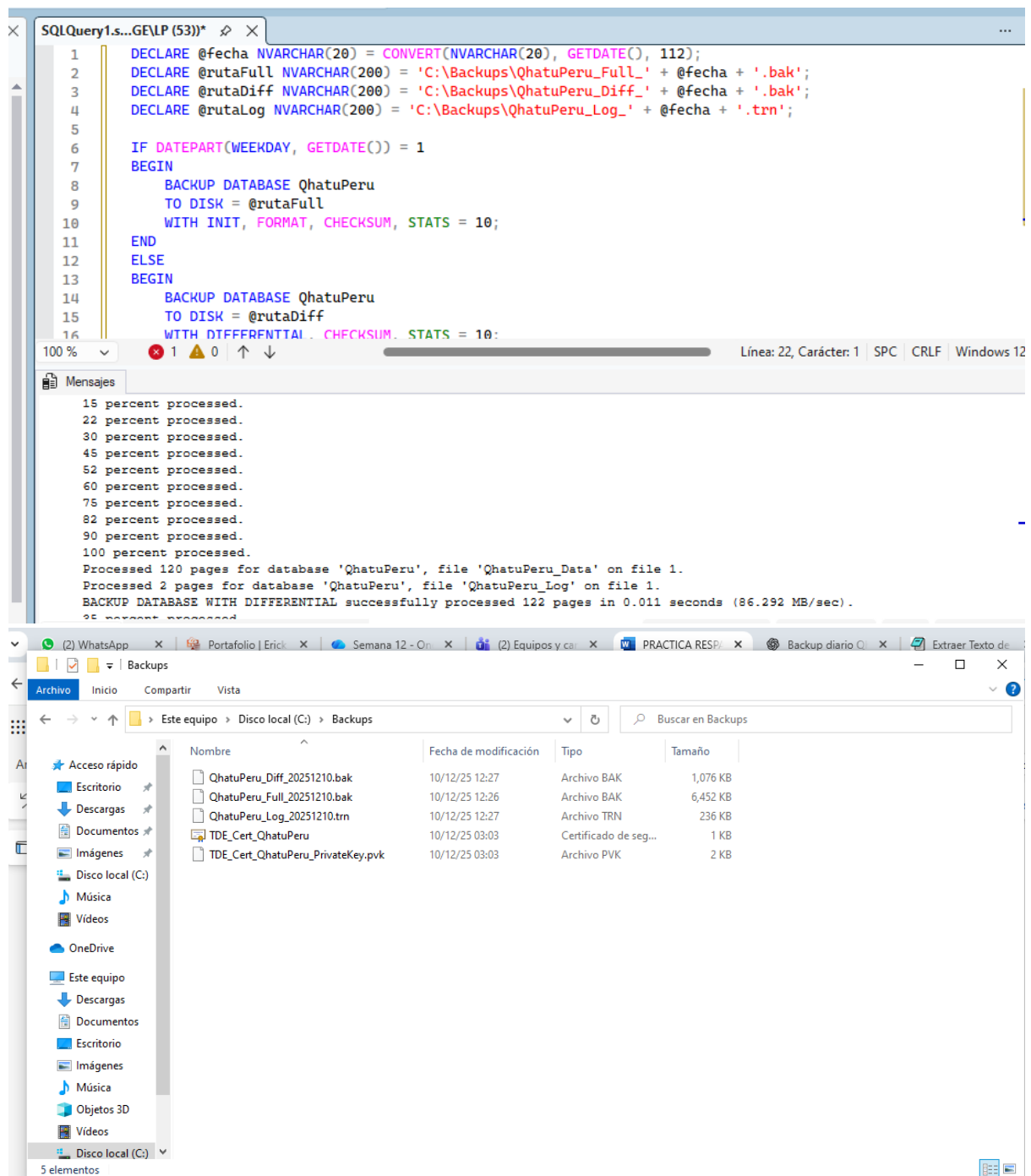
```
    WITH DIFFERENTIAL, CHECKSUM, STATS = 10;
```

```
END
```

```
BACKUP LOG QhatuPeru
```

```
TO DISK = @rutaLog
```

```
WITH CHECKSUM, STATS = 10;
```



3. Justificación técnica

Se combina FULL + DIF + LOG para restauraciones completas o punto-en-el-tiempo. La fecha organiza los backups.

4. Buenas prácticas

- Estrategia combinada balanceada.
- Automatización con SQL Agent.
- Auditoría y verificación recomendadas.

Proyecto 4: Estrategia diferencial: snapshot entre backups completos

1. Enunciado

Automatizar backups diferenciales diarios de lunes a sábado, conservarlos 14 días, limpiar automáticamente archivos antiguos.

2. Script T-SQL

```
-- Declaración de variables DECLARE @fecha NVARCHAR(20); DECLARE @rutaDiff
NVARCHAR(200); DECLARE @fechaLimite DATETIME; DECLARE @file
NVARCHAR(200); DECLARE @cmd NVARCHAR(4000);

-- Asignación de valores SET @fecha = CONVERT(NVARCHAR(20), GETDATE(), 112); SET
@rutaDiff = N'C:\Backups\QhatuPeru_Diff_' + @fecha + N'.bak'; SET @fechaLimite =
DATEADD(DAY, -14, GETDATE());

-- Backup diferencial BACKUP DATABASE QhatuPeru TO DISK = @rutaDiff WITH
DIFFERENTIAL, CHECKSUM, STATS = 10;

-- Cursor para eliminar backups antiguos (>14 días) DECLARE backup_cursor CURSOR
FOR SELECT physical_device_name FROM msdb.dbo.backupmediafamily AS mf JOIN
msdb.dbo.backupset AS bs ON mf.media_set_id = bs.media_set_id WHERE
bs.backup_finish_date < @fechaLimite;

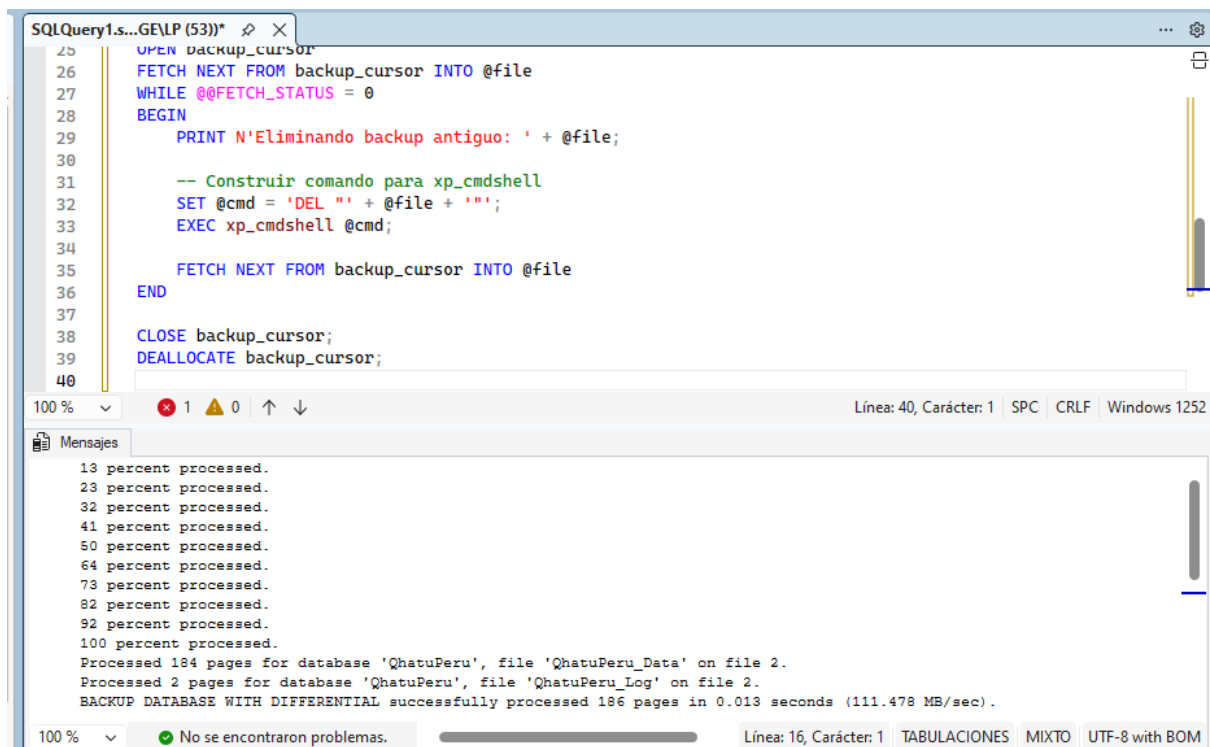
OPEN backup_cursor FETCH NEXT FROM backup_cursor INTO @file WHILE
@@FETCH_STATUS = 0 BEGIN PRINT N'Eliminando backup antiguo: ' + @file;

-- Construir comando para xp_cmdshell
SET @cmd = 'DEL "' + @file + '"';
EXEC xp_cmdshell @cmd;

FETCH NEXT FROM backup_cursor INTO @file

END

CLOSE backup_cursor; DEALLOCATE backup_cursor;
```



The screenshot displays the SQL Server Enterprise Manager interface. The top pane shows a T-SQL script for deleting old backups. The bottom pane shows the execution results, indicating that 100 percent of the backup files were processed successfully.

```
25 OPEN backup_cursor
26 FETCH NEXT FROM backup_cursor INTO @file
27 WHILE @@FETCH_STATUS = 0
28 BEGIN
29     PRINT N'Eliminando backup antiguo: ' + @file;
30
31     -- Construir comando para xp_cmdshell
32     SET @cmd = 'DEL "' + @file + '"';
33     EXEC xp_cmdshell @cmd;
34
35     FETCH NEXT FROM backup_cursor INTO @file
36 END
37
38 CLOSE backup_cursor;
39 DEALLOCATE backup_cursor;
40
```

Mensajes

13 percent processed.
23 percent processed.
32 percent processed.
41 percent processed.
50 percent processed.
64 percent processed.
73 percent processed.
82 percent processed.
92 percent processed.
100 percent processed.
Processed 184 pages for database 'QhatuPeru', file 'QhatuPeru_Data' on file 2.
Processed 2 pages for database 'QhatuPeru', file 'QhatuPeru_Log' on file 2.
BACKUP DATABASE WITH DIFFERENTIAL successfully processed 186 pages in 0.013 seconds (111.478 MB/sec).

3. Justificación técnica

Se mantiene histórico suficiente sin saturar espacio en disco. Diferencial reduce tamaño y tiempo de backup.

4. Buenas prácticas

- Retención limitada (14 días).
- Limpieza automática.
- Nombres de archivo con fecha.

Proyecto 5: Restauración en distintos escenarios

1. Enunciado

Practicar restauraciones en tres escenarios:

- a) Restauración completa.
- b) Restauración punto en el tiempo.
- c) Restauración desde backup alternativo por corrupción.

2. Script T-SQL

```
-- Conectarse a master
```

```
USE master; GO
```

```
-- Poner la base en modo single
```

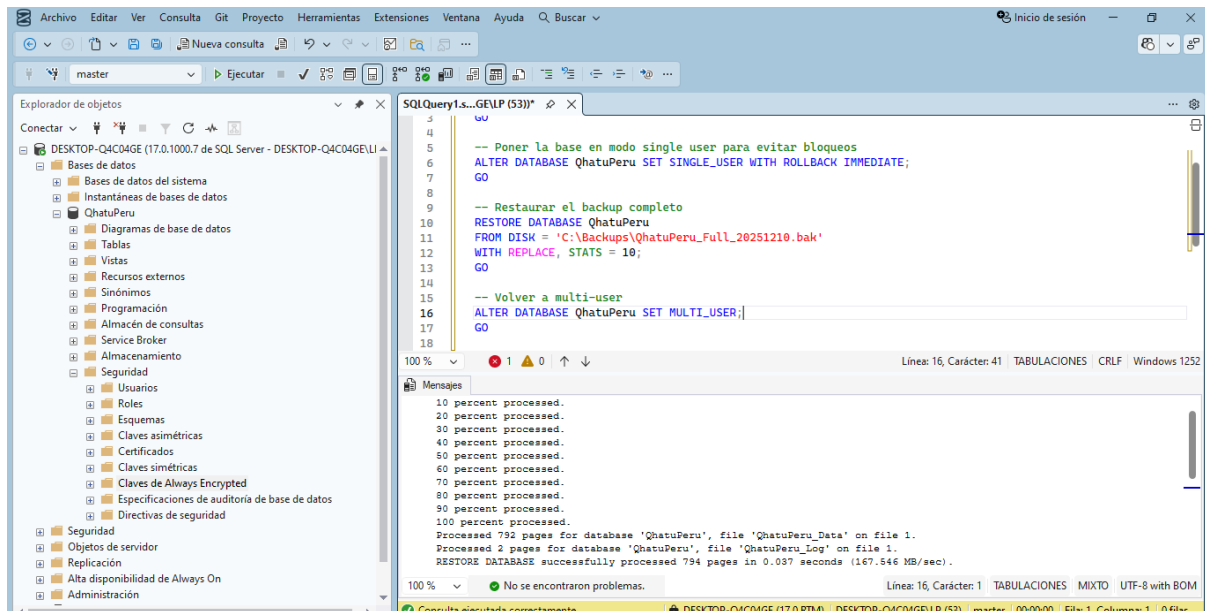
```
user para evitar bloqueos ALTER DATABASE QhatuPeru SET SINGLE_USER  
WITH ROLLBACK IMMEDIATE; GO
```

```
-- Restaurar el backup completo
```

```
RESTORE DATABASE QhatuPeru FROM DISK =  
'C:\Backups\QhatuPeru_Full_20251210.bak' WITH REPLACE, STATS = 10;  
GO
```

```
-- Volver a multi-user
```

```
ALTER DATABASE QhatuPeru SET MULTI_USER; GO
```



```
USE master;
```

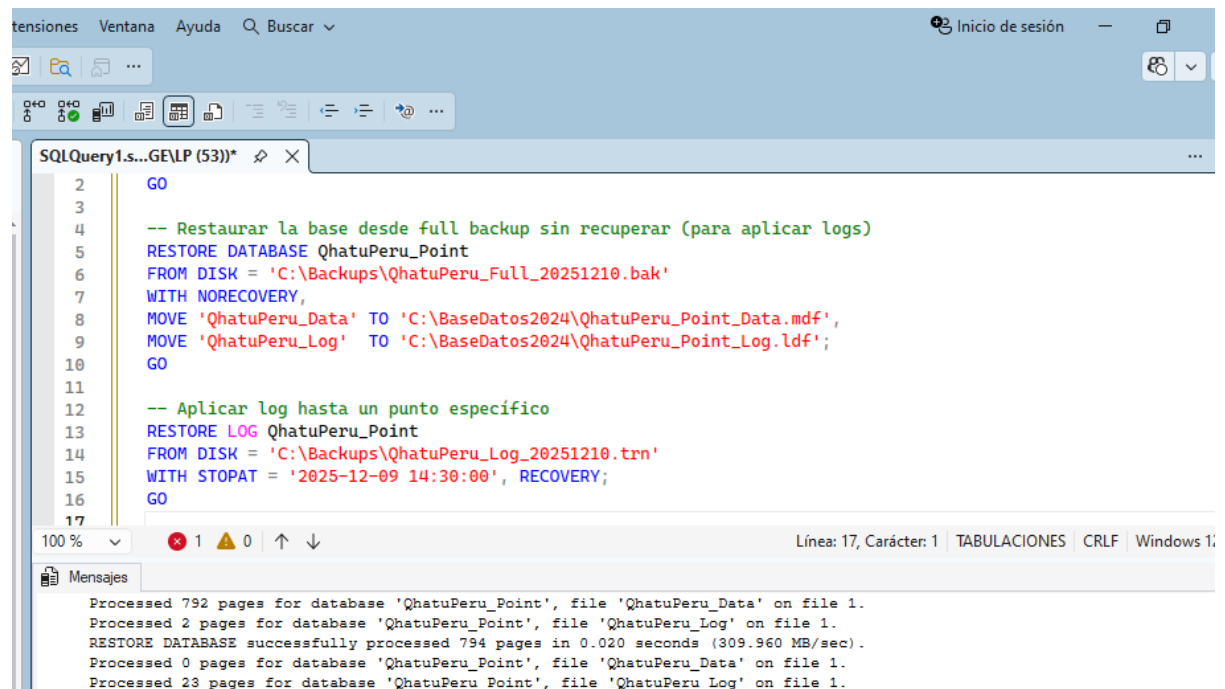
```
GO
```

```
-- Restaurar la base desde full backup sin recuperar (para aplicar  
logs)
```

```
RESTORE DATABASE QhatuPeru_Point  
FROM DISK = 'C:\Backups\QhatuPeru_Full_20251210.bak'  
WITH NORECOVERY,  
MOVE 'QhatuPeru_Data' TO  
'C:\BaseDatos2024\QhatuPeru_Point_Data.mdf',
```

```
MOVE 'QhatuPeru_Log' TO 'C:\BaseDatos2024\QhatuPeru_Point_Log.ldf';  
GO
```

```
-- Aplicar log hasta un punto específico  
RESTORE LOG QhatuPeru_Point  
FROM DISK = 'C:\Backups\QhatuPeru_Log_20251210.trn'  
WITH STOPAT = '2025-12-09 14:30:00', RECOVERY;  
GO
```

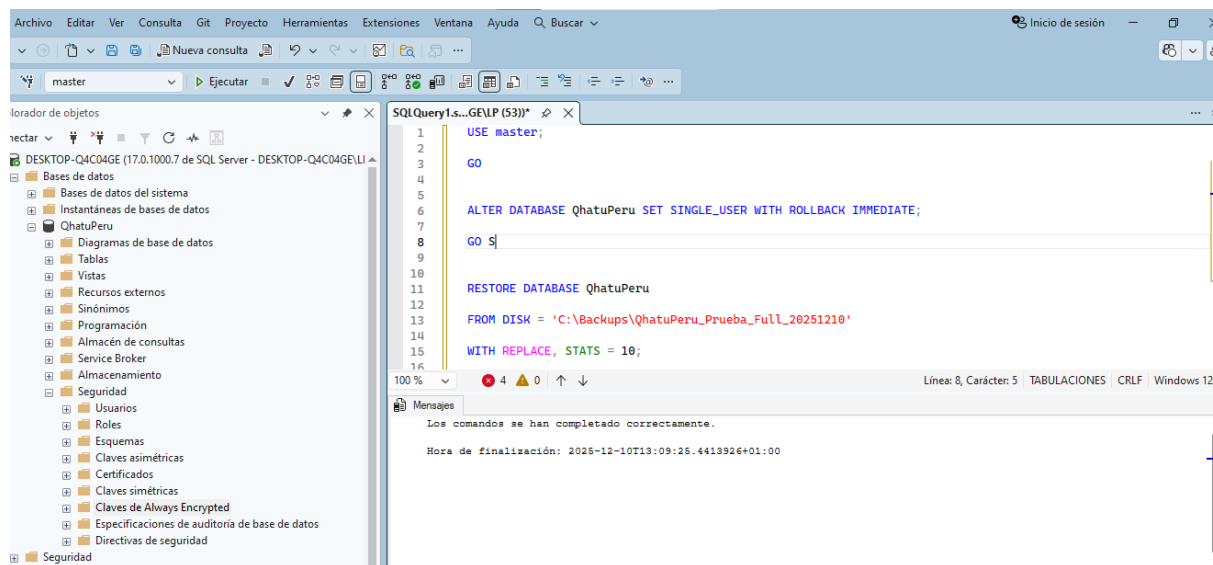


```
-- c) USE master;  
GO
```

```
ALTER DATABASE QhatuPeru SET SINGLE_USER WITH ROLLBACK IMMEDIATE;  
GO
```

```
RESTORE DATABASE QhatuPeru  
FROM DISK = 'C:\Backups\QhatuPeru_Full_Alternate.bak'  
WITH REPLACE, STATS = 10;  
GO
```

```
ALTER DATABASE QhatuPeru SET MULTI_USER;  
GO
```



3. Justificación técnica

Cubren restauración normal, punto en el tiempo y recuperación desde otro backup si el principal falla.

4. Buenas prácticas

- Restauración en base separada para pruebas.
- Uso de NORECOVERY para aplicar logs consecutivos.
- Respaldo alternativo como seguridad extra.

Proyecto 6: Backups de registros de transacciones (LOG) para PITR

1. Enunciado

Cambiar la base a FULL recovery, ejecutar backups de log cada hora y restaurar a un punto en el tiempo.

2. Script T-SQL

```
ALTER DATABASE QhatuPeru SET RECOVERY FULL;
```

```
DECLARE @fecha NVARCHAR(20) = CONVERT(NVARCHAR(20), GETDATE(), 112);  
DECLARE @rutaLog NVARCHAR(200) = 'C:\Backups\QhatuPeru_Log_' +  
@fecha + '.trn';
```

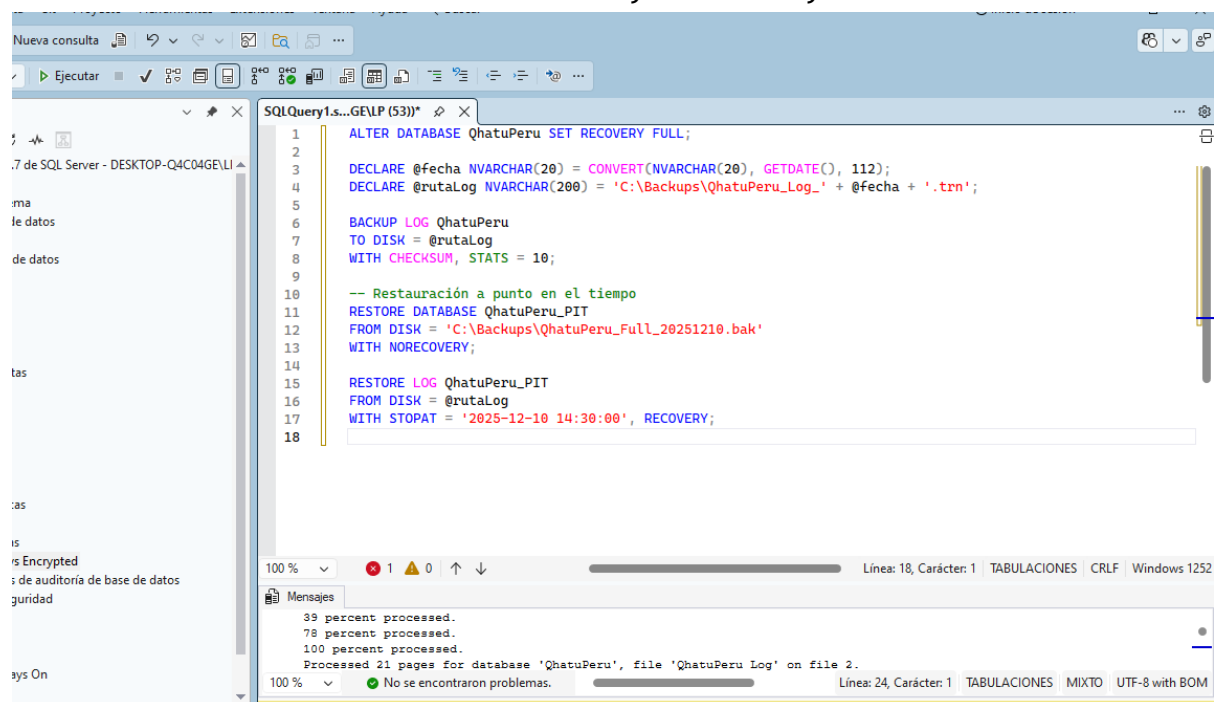
```

BACKUP LOG QhatuPeru
TO DISK = @rutaLog
WITH CHECKSUM, STATS = 10;

-- Restauración a punto en el tiempo
RESTORE DATABASE QhatuPeru_PIT
FROM DISK = 'C:\Backups\QhatuPeru_Full_20251210.bak'
WITH NORECOVERY;

RESTORE LOG QhatuPeru_PIT
FROM DISK = @rutaLog
WITH STOPAT = '2025-12-10 14:30:00', RECOVERY;

```



3. Justificación técnica

Modelo FULL permite restaurar hasta un momento específico, útil ante errores humanos.

4. Buenas prácticas

- Backups de log frecuentes.
- Separación de base restaurada para pruebas.
- Verificación de integridad recomendada.

Proyecto 7: Mantenimiento y verificación de coplas

1. Enunciado

Automatizar verificación periódica de backups, registrar en auditoría y alertar si falla.

2. Script T-SQL

USE QhatuPeru; -- o la base donde quieras registrar los backups

GO

IF OBJECT_ID('BackupAudit', 'U') IS NULL

BEGIN

CREATE TABLE BackupAudit

(

Id INT IDENTITY(1,1) PRIMARY KEY,

TipoBackup NVARCHAR(20),

Fecha DATETIME,

RutaBackup NVARCHAR(200),

Estado NVARCHAR(100)

);

END

GO

USE QhatuPeru; -- asegúrate de estar en la base correcta

GO

DECLARE @ruta NVARCHAR(200) = N'C:\Backups\QhatuPeru_Full_20251210.bak';

DECLARE @estado NVARCHAR(100);

BEGIN TRY

RESTORE VERIFYONLY FROM DISK = @ruta;

SET @estado = N'Éxito';

END TRY

BEGIN CATCH

SET @estado = N'Error: ' + ERROR_MESSAGE();

END CATCH;

INSERT INTO BackupAudit(TipoBackup, Fecha, RutaBackup, Estado)

VALUES(N'FULL', GETDATE(), @ruta, @estado);

```
SQLQuery1.s...GE\LP (53))*
2      GO
3
4      DECLARE @ruta NVARCHAR(200) = N'C:\Backups\QhatuPeru_Full_20251210.bak';
5      DECLARE @estado NVARCHAR(100);
6
7      BEGIN TRY
8          RESTORE VERIFYONLY FROM DISK = @ruta;
9          SET @estado = N'Éxito';
10     END TRY
11     BEGIN CATCH
12         SET @estado = N'Error: ' + ERROR_MESSAGE();
13     END CATCH;
14
15     INSERT INTO BackupAudit(TipoBackup, Fecha, RutaBackup, Estado)
16     VALUES(N'FULL', GETDATE(), @ruta, @estado);
17
```

100 % 5 0 17, Carácter: 1 TABULACIONES CRLF Wi

Mensajes

The backup set on file 1 is valid.

(1 fila afectada)

Hora de finalización: 2025-12-10T13:12:37.8598787+01:00

3. Justificación técnica

Se monitorea el estado de los backups y se registra cualquier fallo para acción inmediata.

4. Buenas prácticas

- Auditoría centralizada.
- Control de errores.
- Prevención proactiva de fallas.

Proyecto 8: Restauraciones en distintos escenarios (corrupción parcial, hardware, pérdida del log)

1. Enunciado

Diseñar restauraciones para:

- A) Archivo de datos corrupto.
- B) Pérdida del log.
- C) Restauración en servidor diferente.

2. Script T-SQL

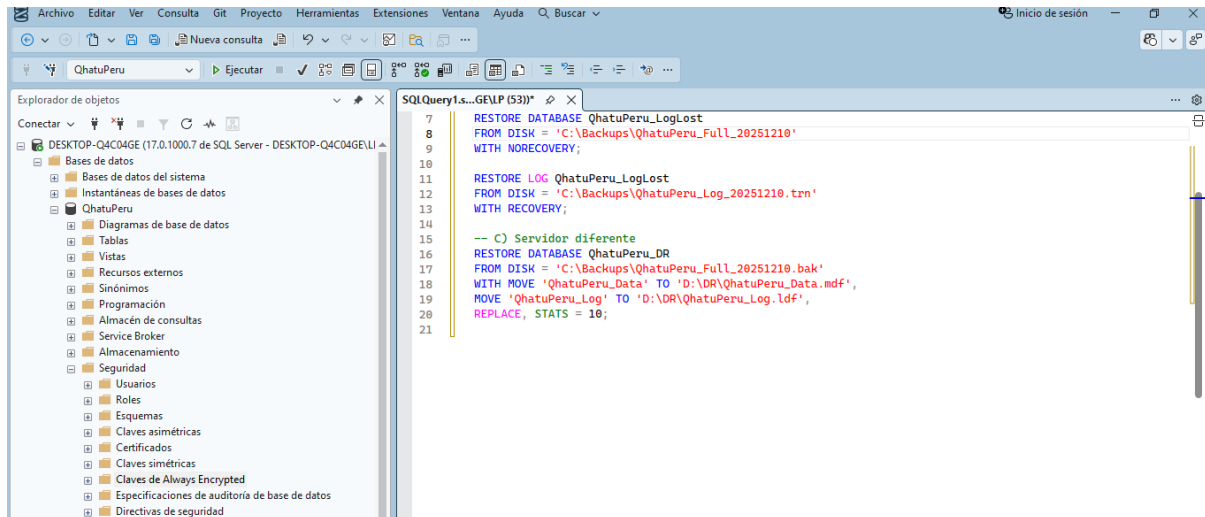
```
-- A) Archivo corrupto
RESTORE DATABASE QhatuPeru
FROM DISK = 'C:\Backups\QhatuPeru_Full_20251210.bak'
WITH REPLACE, STATS = 10;

-- B) Pérdida de log
RESTORE DATABASE QhatuPeru_LogLost
FROM DISK = 'C:\Backups\QhatuPeru_Full_20251210.bak'
WITH NORECOVERY;

RESTORE LOG QhatuPeru_LogLost
FROM DISK = 'C:\Backups\QhatuPeru_Log_20251210.trn'
WITH RECOVERY;

-- C) Servidor diferente
RESTORE DATABASE QhatuPeru_DR
FROM DISK = 'C:\Backups\QhatuPeru_Full_20251210.bak'
WITH MOVE 'QhatuPeru_Data' TO 'D:\DR\QhatuPeru_Data.mdf',
MOVE 'QhatuPeru_Log' TO 'D:\DR\QhatuPeru_Log.ldf',
```

REPLACE, STATS = 10;



3. Justificación técnica

Se cubren escenarios críticos de desastre parcial o migración.

4. Buenas prácticas

- Pruebas en bases separadas.
- Uso de WITH MOVE para cambiar rutas físicas.
- Preparación ante pérdida de logs o corrupción.

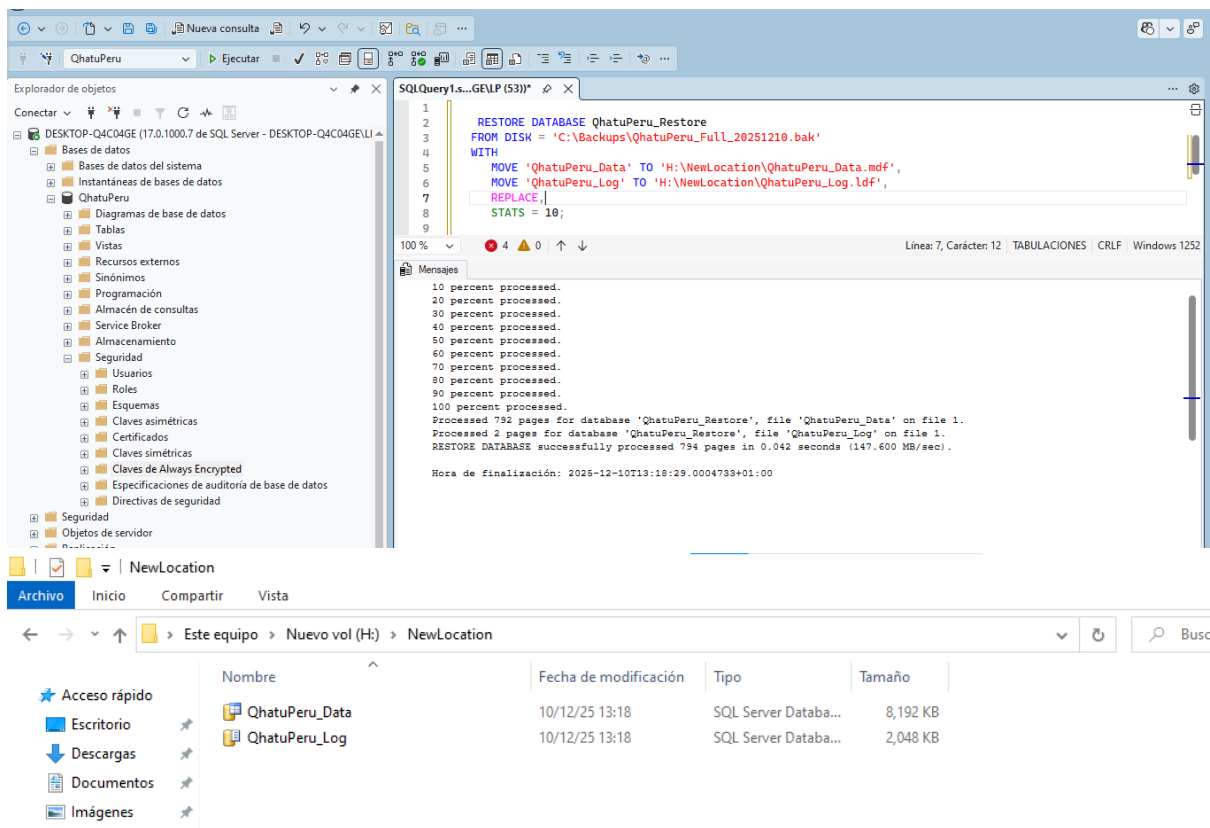
Proyecto 9: Uso de BACKUP DATABASE y RESTORE DATABASE con MOVE

1. Enunciado

Simular mover archivos físicos (mdf/ldf) a otra unidad.

2. Script T-SQL

```
RESTORE DATABASE QhatuPeru_Move
FROM DISK = 'C:\Backups\QhatuPeru_Full_20251210.bak'
WITH MOVE 'QhatuPeru_Data' TO 'H:\NewLocation\QhatuPeru_Data.mdf',
MOVE 'QhatuPeru_Log' TO 'H:\NewLocation\QhatuPeru_Log.ldf',
REPLACE, STATS = 10;
```



3. Justificación técnica

WITH MOVE permite cambiar físicamente los archivos al restaurar, útil por falta de espacio.

4. Buenas prácticas

- Backup previo a restaurar.
- Uso de rutas claras y consistentes.
- Prueba en entorno controlado.

Proyecto 10: Verificación y mantenimiento de backups

1. Enunciado

Crear job que ejecute RESTORE VERIFYONLY para todos los .bak en carpeta y compare con msdb.backupset, registrar en BackupAudit.

2. Script T-SQL

```
DECLARE @file NVARCHAR(200);
DECLARE @estado VARCHAR(50);
```


4. Buenas prácticas

- Auditoría centralizada.
- Cursor para iterar múltiples archivos.
- Registro detallado de estado.

do backups completos y logs.

4. Buenas prácticas

- Plan DR documentado.
- Prueba de restauración completa.
- Registro de tiempos y lecciones aprendidas.

Proyecto 11: Plan de mantenimiento – limpiezas y retención usando msdb

1. Enunciado

Crear plan de mantenimiento que elimine históricos de backup antiguos (sp_delete_backuphistory) y limpie archivos físicos mayores a X días.

2. Script T-SQL

```
USE QhatuPeru; GO
```

```
-- 1. Parámetros de retención DECLARE @RetentionDays INT = 7; --  
Cambia según tu política de retención DECLARE @OldestDate DATETIME;
```

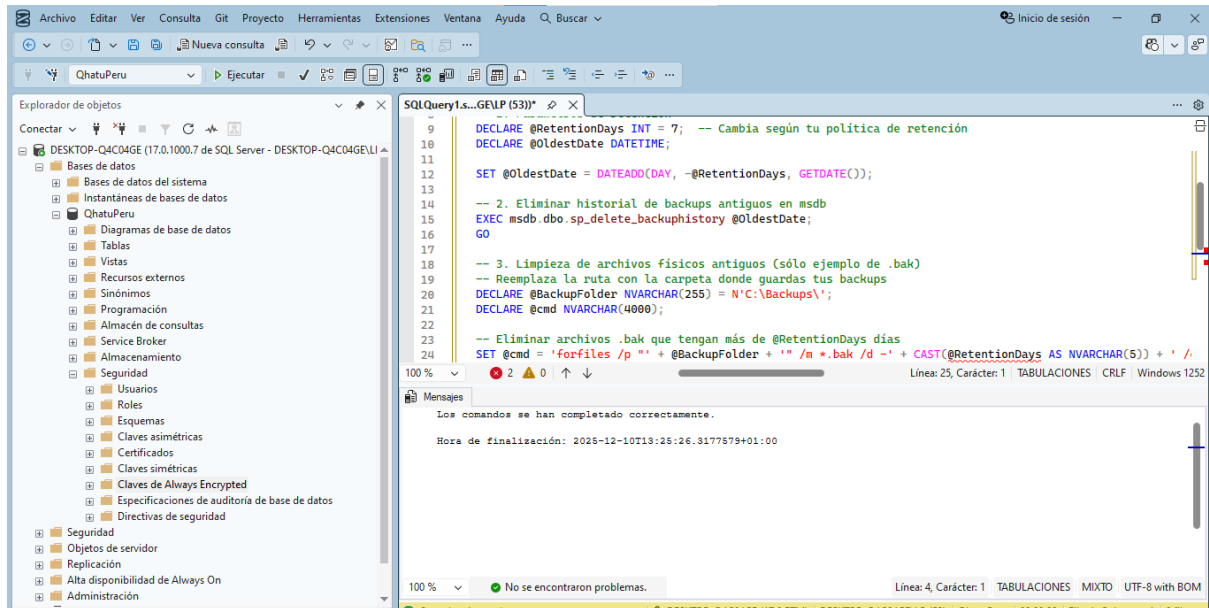
```
SET @OldestDate = DATEADD(DAY, -@RetentionDays, GETDATE());
```

```
-- 2. Eliminar historial de backups antiguos en msdb EXEC  
msdb.dbo.sp_delete_backuphistory @OldestDate; GO
```

```
-- 3. Limpieza de archivos físicos antiguos (sólo ejemplo de .bak) -  
- Reemplaza la ruta con la carpeta donde guardas tus backups DECLARE  
@BackupFolder NVARCHAR(255) = N'C:\Backups'; DECLARE @cmd  
NVARCHAR(4000);
```

```
-- Eliminar archivos .bak que tengan más de @RetentionDays días SET  
@cmd = 'forfiles /p "' + @BackupFolder + '" /m *.bak /d -' +  
CAST(@RetentionDays AS NVARCHAR(5)) + ' /c "cmd /c del @path"';
```

-- Ejecutar el comando EXEC xp_cmdshell @cmd; GO



3. Justificación técnica

Evita saturación de msdb y del disco, manteniendo únicamente backups recientes.

4. Buenas prácticas

- Retención limitada.
- Limpieza automatizada.
- Uso de `sp_delete_backuphistory` para mantener integridad del sistema.

Proyecto 12: Uso avanzado de BACKUP y RESTORE con opciones

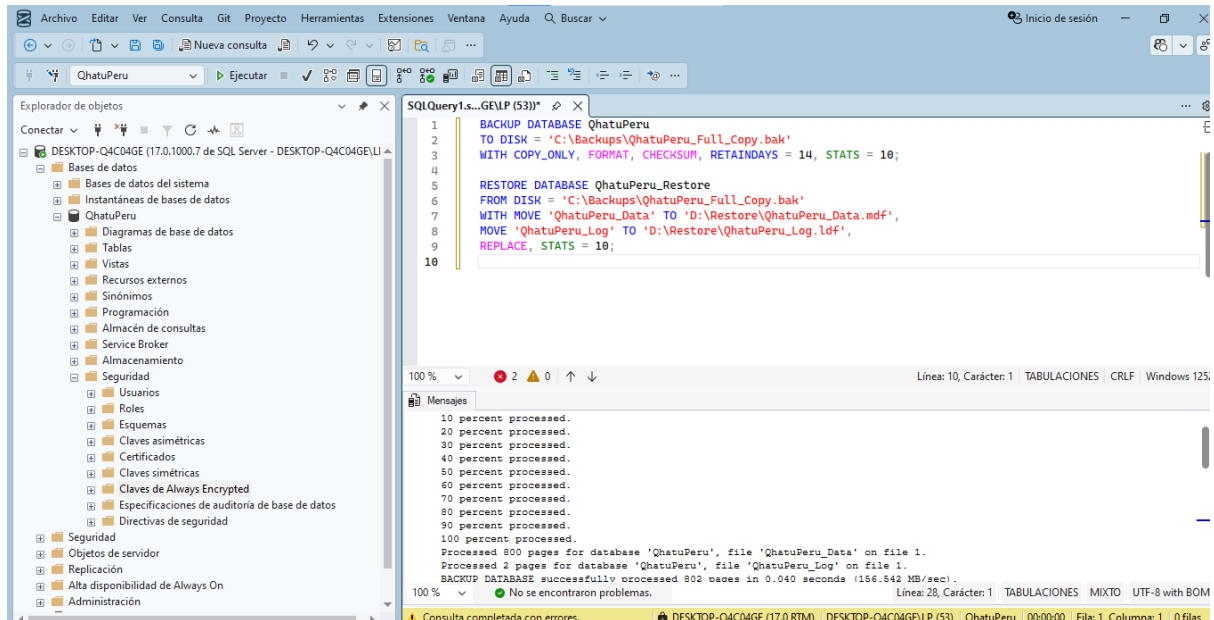
1. Enunciado

Practicar opciones avanzadas: COPY_ONLY, FORMAT, WITH CHECKSUM, WITH RETAINDAYS, WITH STATS, WITH MOVE, WITH REPLACE.

2. Script T-SQL

```
BACKUP DATABASE QhatuPeru
TO DISK = 'C:\Backups\QhatuPeru_Full_Copy.bak'
WITH COPY_ONLY, FORMAT, CHECKSUM, RETAINDAYS = 14, STATS = 10;
```

```
RESTORE DATABASE QhatuPeru_Restore
FROM DISK = 'C:\Backups\QhatuPeru_Full_Copy.bak'
WITH MOVE 'QhatuPeru_Data' TO 'D:\Restore\QhatuPeru_Data.mdf',
MOVE 'QhatuPeru_Log' TO 'D:\Restore\QhatuPeru_Log.ldf',
REPLACE, STATS = 10;
```



3. Justificación técnica

COPY_ONLY permite backup sin afectar cadena de backups diferenciales; RETAINDAYS define retención mínima; CHECKSUM valida integridad.

4. Buenas prácticas

- Prevención de corrupción.
- Documentación de cada opción.
- Uso de ruta separada para pruebas.

Proyecto 13: Restauración punto en el tiempo (recuperar fila borrada)

1. Enunciado

Simular borrado de filas en tabla ARTICULO y recuperarlas mediante restauración a base temporal.

2. Script T-SQL

```
-- Supongamos que se borraron filas
DELETE FROM ARTICULO WHERE CodArticulo = 1;

-- Restaurar backup a base temporal
RESTORE DATABASE QhatuPeru_Temp
FROM DISK = 'C:\Backups\QhatuPeru_Full_20251210.bak'
WITH NORECOVERY;

RESTORE LOG QhatuPeru_Temp
FROM DISK = 'C:\Backups\QhatuPeru_Log_20251210.trn'
WITH STOPAT = '2025-12-10 10:00:00', RECOVERY;

-- Extraer los datos borrados y reinsertar en original
INSERT INTO ARTICULO
SELECT * FROM QhatuPeru_Temp.dbo.ARTICULO
WHERE CodArticulo = 1;
```

3. Justificación técnica

Se restaura a otra base para extraer sólo los datos perdidos sin afectar producción.

4. Buenas prácticas

- Restauración en base temporal.
- PITR para errores humanos.
- Inserción controlada de datos recuperados.

Proyecto 14: Planes de mantenimiento – chequeos de integridad y tareas automáticas

1. Enunciado

Ejecutar semanalmente: DBCC CHECKDB, reconstrucción de índices fragmentados >30%, actualización de estadísticas y limpieza de backups antiguos.

2. Script T-SQL

```
-- Verificación de integridad
DBCC CHECKDB('QhatuPeru') WITH NO_INFOMSGS, ALL_ERRORMSG;
```

```
-- Reconstrucción índices fragmentados
DECLARE @sql NVARCHAR(MAX) = '';
SELECT @sql += 'ALTER INDEX ALL ON [' + t.name + '] REBUILD;'
FROM sys.tables t
JOIN sys.indexes i ON t.object_id = i.object_id
WHERE i.type_desc = 'CLUSTERED' AND i.avg_fragmentation_in_percent >
30;

EXEC sp_executesql @sql;

-- Actualización de estadísticas
EXEC sp_updatestats;

-- Limpieza de backups antiguos (>30 días)
EXEC msdb.dbo.sp_delete_backuphistory @oldest_date = DATEADD(DAY, -
30, GETDATE());
```

3. Justificación técnica

Mantiene integridad, performance de índices y estadísticas actualizadas para optimizar consultas.

4. Buenas prácticas

- Chequeo periódico.
- Reconstrucción selectiva de índices.
- Limpieza de historial de backups.

Proyecto 15: Políticas de retención y compresión de backups

1. Enunciado

Implementar backups con compresión y retención: FULL 4 semanas, DIF 7 días, LOG 48 horas.

2. Script T-SQL

```
DECLARE @fecha NVARCHAR(20) = CONVERT(NVARCHAR(20), GETDATE(), 112);
DECLARE @rutaFull NVARCHAR(200) = 'C:\Backups\QhatuPeru_Full_' +
@fecha + '.bak';
```

```

BACKUP DATABASE QhatuPeru
TO DISK = @rutaFull
WITH COMPRESSION, CHECKSUM, STATS = 10;

-- Limpieza
EXEC xp_cmdshell 'forfiles /p "C:\Backups" /s /m *.bak /d -28 /c
"cmd /c del @path"';
EXEC xp_cmdshell 'forfiles /p "C:\Backups" /s /m *.trn /d -2 /c "cmd
/c del @path"';

```

3. Justificación técnica

Optimiza espacio y asegura política de retención automática.

4. Buenas prácticas

- Compresión de backups.
- Retención clara y automática.
- Uso de forfiles para limpieza eficiente.

Proyecto 16: Políticas de retención de backups – Implementación y reporte

1. Enunciado

Diseñar política: Full 30 días, Diferencial 14 días, Log 7 días; generar reporte automático de cumplimiento.

2. Script T-SQL

```

SELECT
    bs.database_name,
    bs.backup_start_date,
    bs.backup_finish_date,
    CASE
        WHEN bs.type = 'D' AND bs.backup_finish_date >= DATEADD(DAY,
-30, GETDATE()) THEN 'Cumple'
        WHEN bs.type = 'I' AND bs.backup_finish_date >= DATEADD(DAY,
-14, GETDATE()) THEN 'Cumple'
        WHEN bs.type = 'L' AND bs.backup_finish_date >= DATEADD(DAY,

```

```

-7, GETDATE()) THEN 'Cumple'
      ELSE 'No Cumple'
    END AS EstadoRetencion
FROM msdb.dbo.backupset bs
WHERE bs.database_name = 'QhatuPeru'
ORDER BY bs.backup_finish_date DESC;

```

3. Justificación técnica

Permite monitorear cumplimiento de política y tomar medidas si hay backups fuera de retención.

4. Buenas prácticas

- Reporte automático.
- Seguimiento de cumplimiento.
- Facilita auditoría interna.

Proyecto 17: Verificación automática post-backup y notificación (simulada)

1. Enunciado

Backup, verificación con RESTORE VERIFYONLY, registrar resultado y simular envío de correo si falla.

2. Script T-SQL

```

DECLARE @ruta NVARCHAR(200) = 'C:\Backups\QhatuPeru_Full_' +
  CONVERT(NVARCHAR(20), GETDATE(), 112) + '.bak';
DECLARE @estado VARCHAR(50);

BEGIN TRY
  BACKUP DATABASE QhatuPeru
  TO DISK = @ruta
  WITH COMPRESSION, CHECKSUM, STATS = 10;

  RESTORE VERIFYONLY FROM DISK = @ruta;
  SET @estado = 'Éxito';
END TRY
BEGIN CATCH

```

```

SET @estado = 'Error: ' + ERROR_MESSAGE();
-- Simulación envío correo: insert en tabla
INSERT INTO BackupAudit(TipoBackup, Fecha, RutaBackup, Estado)
VALUES('FULL', GETDATE(), @ruta, 'Notificación enviada: ' +
@estado);
END CATCH;

INSERT INTO BackupAudit(TipoBackup, Fecha, RutaBackup, Estado)
VALUES('FULL', GETDATE(), @ruta, @estado);

```

3. Justificación técnica

Detecta fallos automáticamente y registra acción correctiva (simulada).

4. Buenas prácticas

- Registro automático de errores.
- Simulación de alerta sin depender de correo real.
- Verificación post-backup.

Proyecto 18: Simulacro de desastre y recuperación completa (DR drill)

1. Enunciado

Simular pérdida total del servidor y restaurar en servidor alternativo usando backups disponibles.

2. Script T-SQL

```

-- Restaurar en servidor alternativo
RESTORE DATABASE QhatuPeru_DR
FROM DISK = 'C:\Backups\QhatuPeru_Full_20251210.bak'
WITH MOVE 'QhatuPeru_Data' TO 'D:\DR\QhatuPeru_Data.mdf',
MOVE 'QhatuPeru_Log' TO 'D:\DR\QhatuPeru_Log.ldf',
NORECOVERY, STATS = 10;

-- Restaurar DIF y LOG
RESTORE DATABASE QhatuPeru_DR
FROM DISK = 'C:\Backups\QhatuPeru_Diff_20251210.bak'

```

```
WITH NORECOVERY, STATS = 10;
```

```
RESTORE LOG QhatuPeru_DR  
FROM DISK = 'C:\Backups\QhatuPeru_Log_20251210.trn'  
WITH RECOVERY, STATS = 10;
```

3. Justificación técnica

Simula un desastre total y verifica la recuperación usando backups completos y logs.

4. Buenas prácticas

- Plan DR documentado.
- Prueba de restauración completa.
- Registro de tiempos y lecciones aprendidas.

Proyecto 19: Simulación de recuperación DR con copia a ubicación remota

1. Enunciado

Generar backups, copiarlos a ubicación remota, y restaurar desde esa ubicación.

2. Script T-SQL

```
-- Copiar backup a carpeta remota (simulado)  
EXEC xp_cmdshell 'COPY "C:\Backups\QhatuPeru_Full_20251210.bak"  
"\\RemoteServer\Backups"';  
  
-- Restaurar desde ubicación remota  
RESTORE DATABASE QhatuPeru_DR_Remote  
FROM DISK = '\\RemoteServer\Backups\QhatuPeru_Full_20251210.bak'  
WITH MOVE 'QhatuPeru_Data' TO 'D:\DR_Remote\QhatuPeru_Data.mdf',  
MOVE 'QhatuPeru_Log' TO 'D:\DR_Remote\QhatuPeru_Log.ldf',  
REPLACE, STATS = 10;
```

3. Justificación técnica

Asegura continuidad ante falla local, usando backups remotos.

4. Buenas prácticas

- Ubicación remota separada.
- Control de rutas físicas con WITH MOVE.
- Copia segura de archivos.

Proyecto 20: Automatización completa – script maestro

1. Enunciado

Script único que detecte último FULL, determine si toca FULL o DIF, ejecute backups, logs, verifique e inserte reporte.

2. Script T-SQL

```

DECLARE @ultimoFull DATETIME;
SELECT @ultimoFull = MAX(backup_finish_date)
FROM msdb.dbo.backupset
WHERE database_name = 'QhatuPeru' AND type = 'D';

DECLARE @ruta NVARCHAR(200) = 'C:\Backups\QhatuPeru_';
DECLARE @fecha NVARCHAR(20) = CONVERT(NVARCHAR(20), GETDATE(), 112);

IF @ultimoFull IS NULL OR DATEDIFF(DAY, @ultimoFull, GETDATE()) >= 7
BEGIN
    SET @ruta = @ruta + 'Full_' + @fecha + '.bak';
    BACKUP DATABASE QhatuPeru TO DISK = @ruta WITH FORMAT, CHECKSUM,
STATS = 10;
END
ELSE
BEGIN
    SET @ruta = @ruta + 'Diff_' + @fecha + '.bak';
    BACKUP DATABASE QhatuPeru TO DISK = @ruta WITH DIFFERENTIAL,
CHECKSUM, STATS = 10;
END

-- Backup log
DECLARE @rutaLog NVARCHAR(200) = 'C:\Backups\QhatuPeru_Log_' +
@fecha + '.trn';
BACKUP LOG QhatuPeru TO DISK = @rutaLog WITH CHECKSUM, STATS = 10;

-- Verificación y reporte

```

```
DECLARE @estado VARCHAR(50);
BEGIN TRY
    RESTORE VERIFYONLY FROM DISK = @ruta;
    SET @estado = 'Éxito';
END TRY
BEGIN CATCH
    SET @estado = 'Error: ' + ERROR_MESSAGE();
END CATCH;

INSERT INTO BackupAudit(TipoBackup, Fecha, RutaBackup, Estado)
VALUES('Automatizado', GETDATE(), @ruta, @estado);
```

3. Justificación técnica

Automatiza decisión FULL vs DIF, realiza backup de logs y verifica integridad en un solo flujo.

4. Buenas prácticas

- Script maestro para todo tipo de backup.
- Automatización completa.
- Auditoría centralizada en BackupAudit.
- Control de errores y verificación.