Exploring SQL Server Backup Types

Part 1: Research – SQL Server Backup Types

1. Full Backup

- **When Used**: As a base for all other backups. Typically scheduled weekly.
- **What It Includes**: Entire database including system tables and user data.
- **Pros**:
    - Simplifies restore
    - Single file contains complete data
- **Cons**:
    - Time-consuming for large databases
    - High storage usage
- **Real-World Scenario**: A **banking system** uses full backups every Sunday to ensure complete data recovery baseline.

2. Differential Backup

- **When Used**: Mid-week or daily after a full backup.
- **What It Includes**: All data changed since the last full backup.
- **Pros**:
    - Faster than full backup
    - Requires less space
- **Cons**:
    - Dependent on last full backup
    - Multiple files needed during recovery
- **Real-World Scenario**: An **e-learning platform** uses nightly differential backups to protect progress and submissions.

3. Transaction Log Backup

- **When Used**: In FULL recovery model, taken frequently (e.g., hourly).
- **What It Includes**: All transactions since last log backup.
- **Pros**:
    - Enables point-in-time recovery
    - Small in size
- **Cons**:
    - Requires careful restore sequencing
    - Not possible in SIMPLE recovery model
- **Real-World Scenario**: A **ticketing system** uses transaction log backups every hour to prevent data loss during high-traffic events.

4. Copy-Only Backup

- **When Used**: For ad-hoc or one-off backups without affecting the backup chain.
- **What It Includes**: Same as a full or log backup, but doesn't affect backup sequence.
- **Pros**:
    - Safe for manual backups
    - Doesn't interfere with scheduled backups
- **Cons**:
    - Cannot be used as part of differential or log chain
- **Real-World Scenario**: A **development team** copies production data for testing without disrupting the regular backup chain.

5. File/Filegroup Backup

- **When Used**: For very large databases using multiple filegroups.
- **What It Includes**: Selected files or filegroups.
- **Pros**:
    - Optimized for large DBs
    - Enables partial restores

- **Cons**:
    - Complex management
    - Recovery across filegroups must be consistent
- **Real-World Scenario**: A **research database** with petabytes of genomic data stored in separate filegroups.

Step 1: Create Test Database

```
-- Step 1: Create Test Database
CREATE DATABASE TrainingDB;
GO
USE TrainingDB;
GO
CREATE TABLE Students (
    StudentID INT PRIMARY KEY,
    FullName NVARCHAR(100),
    EnrollmentDate DATE
);
INSERT INTO Students VALUES
(1, 'Sara Ali', '2023-09-01'),
(2, 'Mohammed Nasser', '2023-10-15');
```

DESKTOP-E29KCG6.T...DB - dbo.Students | SQLQuery33.sql - D..

	StudentID	FullName	EnrollmentDate
	1	Sara Ali	2023-09-01
	2	Mohammed N...	2023-10-15
▶*	NULL	NULL	NULL

Step 2: Perform Backup Operations

```
-- Step 2.1: Full Backup
BACKUP DATABASE TrainingDB TO DISK = 'C:\Program Files\Microsoft SQL
Server\MSSQL16.MSSQLSERVER\MSSQL\Backup\TrainingDB_Full1.bak';
```

```
-- Step 2.1: Full Backup
BACKUP DATABASE TrainingDB TO DISK = 'C:\Program Files\Microsoft SQL Server\MSSQL16.MSSQLSERVER\MSSQL\Backup\TrainingDB_Full1.bak';
```

-- Step 2.2: Insert New Data
INSERT INTO Students VALUES (3, 'Fatma Said', '2024-01-10');

DESKTOP-E29KCG6.T...DB - dbo.Students | SQLQuery33.sql - D...G6\

StudentID	FullName	EnrollmentDate
1	Sara Ali	2023-09-01
2	Mohammed N...	2023-10-15
3	Fatma Said	2024-01-10
NULL	NULL	NULL

-- Step 2.3: Differential Backup
BACKUP DATABASE TrainingDB TO DISK = 'C:\Program Files\Microsoft SQL Server\MSSQL16.MSSQLSERVER\MSSQL\Backup\TrainingDB_Diff.bak' WITH DIFFERENTIAL;

```
-- Step 2.3: Differential Backup
BACKUP DATABASE TrainingDB TO DISK = 'C:\Program Files\Microsoft SQL Server\MSSQL16.MSSQLSERVER\MSSQL\Backup\TrainingDB_Diff.bak' WITH DIFFERENTIAL;
```

-- Step 2.4: Transaction Log Backup
ALTER DATABASE TrainingDB SET RECOVERY FULL;
BACKUP LOG TrainingDB TO DISK = 'C:\Program Files\Microsoft SQL Server\MSSQL16.MSSQLSERVER\MSSQL\Backup\TrainingDB_Log.trn';

```
-- Step 2.4: Transaction Log Backup
ALTER DATABASE TrainingDB SET RECOVERY FULL;
BACKUP LOG TrainingDB TO DISK = 'C:\Program Files\Microsoft SQL Server\MSSQL16.MSSQLSERVER\MSSQL\Backup\TrainingDB_Log.trn';
```

-- Step 2.5: Copy-Only Backup
BACKUP DATABASE TrainingDB TO DISK = 'C:\Program Files\Microsoft SQL Server\MSSQL16.MSSQLSERVER\MSSQL\Backup\TrainingDB_CopyOnly.bak' WITH COPY_ONLY;

```
-- Step 2.5: Copy-Only Backup
BACKUP DATABASE TrainingDB TO DISK = 'C:\Program Files\Microsoft SQL Server\MSSQL16.MSSQLSERVER\MSSQL\Backup\TrainingDB_CopyOnly.bak' WITH COPY_ONLY;
```

Part 3: Real-World Scenario Simulation

Backup Frequency:

- **Full Backup**: Every Sunday @ 2:00 AM
- **Differential Backup**: Every Sunday @ 2:00 AM
- **Transaction Log Backup**: Every hour (24x7)

HospitalDB Backup Plan	
Backup Frequency:	• **Full Backup**: Every Sunday at 2:00 AM • **Differential Backup**: Every Sunday at 2:00 AM • **Transaction Log Backup**: Every hour (24x7)
Folder Structure & Naming Convention:	• **Full:** 'C:\Program Files\Microsoft SQL Server\MSSQL16.MSSQLSERVER\MSSQL\Backup\HospitalDB_Full_YYYYMMDD.bak • **Differential:** 'C:\Program Files\Microsoft SQL Server\MSSQL16.MSSQLSERVER\MSSQL\Backup\HospitalDB_Diff_YYYYMMDD.bak • **Log:** 'C:\Program Files\Microsoft SQL Server\MSSQL16.MSSQLSERVER\MSSQL\Backup\HospitalDB_Log_YYYYMMDD_HH.trn

```
ALTER DATABASE HospitalDB SET RECOVERY FULL;

-- Weekly Full Backup (Sunday)
BACKUP DATABASE HospitalDB
TO DISK = 'C:\HospitalBackups\Full\HospitalDB_Full_20250601.bak'
WITH INIT, COMPRESSION, NAME = 'Weekly Full Backup';

-- Daily Differential Backup (Mon–Sat)
BACKUP DATABASE HospitalDB
TO DISK = 'C:\HospitalBackups\Differential\HospitalDB_Diff_20250602.bak'
WITH DIFFERENTIAL, INIT, COMPRESSION, NAME = 'Nightly Diff Backup';

-- Hourly Transaction Log Backup
BACKUP LOG HospitalDB
TO DISK = 'C:\HospitalBackups\Logs\HospitalDB_Log_20250602_08.trn'
WITH INIT, COMPRESSION, NAME = 'Hourly Log Backup';
```