What is tempdb Database

tempdb System Database

- The tempdb system database is a global resource that's available to all users connected to the instance of SQL Server.
- **Temporary user objects that are explicitly created.** They include global or local temporary tables and indexes, temporary stored procedures, table variables, tables returned in table-valued functions, and cursors.
- **Internal objects that the database engine creates.** They include:
- Work tables to store intermediate results for spools, cursors, sorts, and temporary large object (LOB) storage.
- Work files for hash join or hash aggregate operations.
- Intermediate sort results for operations such as creating or rebuilding indexes (if SORT_IN_TEMPDB is specified), or certain GROUP BY, ORDER BY, or UNION queries.
- **Version stores,** which are collections of data pages that hold the data rows that support features for row versioning.

Physical Properties of master database

- The following table lists the initial configuration values of the **tempdb** data and log files for SQL Server SQL Managed Instance.
- The sizes of these files may vary slightly for different editions of SQL Server.

File	Logical Name	Physical Name	Initial Size	File Growth
Primary data	tempdev	tempdb.mdf	8 MB	Autogrow by 64 MB until the disk is full.
Secondary data files	temp#	tempdb_mssql_#.ndf	8 MB	Autogrow by 64 MB until the disk is full.
Log	mastlog	mastlog.ldf		Autogrow by 64 MB to a maximum of 2 terabytes.

Physical Properties of master database (contd..)

- The number of secondary data files depends on the number of (logical) processors on the machine.
- As a general rule, if the number of logical processors is less than or equal to eight, use the same number of data files as logical processors.
- If the number of logical processors is greater than eight, use eight data files.
- Then if contention continues, increase the number of data files by multiples of four until the contention decreases to acceptable levels, or make changes to the workload/code.

Restrictions of tempdb database

The following operations cannot be performed on the **master** database:

- Adding filegroups.
- Backing up or restoring the database.
- Changing collation. The default collation is the server collation.
- Changing the database owner. tempdb is owned by sa.
- Creating a database snapshot.
- Dropping the database.
- Dropping the guest user from the database.
- Enabling Change Data Capture.
- Participating in database mirroring.

Restrictions of tempdb database (contd)

- Removing the primary filegroup, primary data file, or log file.
- Renaming the database or primary filegroup.
- Running DBCC CHECKALLOC.
- Running DBCC CHECKCATALOG.
- Setting the database to OFFLINE.
- Setting the database or primary filegroup to READ_ONLY.

Optimizing tempdb performance in SQL Server

- The size and physical placement of the tempdb database can affect the performance of a system.
- If possible, use instant file initialization to improve the performance of growth operations for data files.
- Preallocate space for all tempdb files by setting the file size to a value large enough to accommodate the typical workload in the environment.
- Data files should be of equal size within each filegroup, because SQL Server uses a proportional-fill algorithm that favors allocations in files with more free space.
- Set the file growth increment to a reasonable size to prevent the tempdb database files from growing by too small a value.
- Put the tempdb database on a fast I/O subsystem.
- Put the tempdb database on disks that differ from the disks that user databases use.

Moving the tempdb database

- Determine the logical file names of the tempdb database and their current location on the disk.
- Change the location of each file by using ALTER DATABASE.
- Stop and restart the instance of SQL Server.
- Verify the file change.
- Delete the tempdb.mdf and templog.ldf files from the original location.