

[Server Info](#)
[Hardware Info](#)
[Database Info](#)
[Recovery Model](#)
[Backup](#)
[Backup Device](#)
[Restore](#)
[SQL Agent Jobs](#)
[Training Info](#)

Server Information

Find SQL Server Version

```
SELECT @@VERSION
```

Find SQL Server Version Information

```
xp_msver
```

Find Server Properties

```
SELECT SERVERPROPERTY('productversion') AS 'Product Version'  
      ,SERVERPROPERTY ('productlevel') AS 'Service Pack'  
      ,SERVERPROPERTY ('edition') AS 'Edition'  
      ,@@SERVERNAME AS [@@SERVERNAME]  
      ,CAST(SERVERPROPERTY('MACHINENAME') AS VARCHAR(128)) + COALESCE('', +  
        CAST(SERVERPROPERTY('INSTANCENAME') AS VARCHAR(128)), '') As  
        'Instance Name'
```

Reference: <https://msdn.microsoft.com/en-us/library/ms174396.aspx>

Find Port Number for SQL Server Instances

If you have more than one SQL instance on the server you'll need to find the port number SQL Server is listening on. The default instance will be listening on port 1433.

```
sp_readerrorlog 0, 1, N'Server is listening on'  
GO
```

Reference: <https://social.msdn.microsoft.com/Forums/sqlserver/en-US/09b69020-cbab-4fe0-a5e2-2f4865dbdd85/spreaderrorlog?forum=sqlgetstarted>

[Top](#)

Find Account Running SQL Server Agent

In SQL 2005/2008

```
DECLARE @ServiceAccount NVARCHAR(128);  
EXEC master.dbo.xp_regread  
      'HKEY_LOCAL_MACHINE',  
      'SYSTEM\CurrentControlSet\services\SQLSERVERAGENT',  
      'ObjectName',  
      @ServiceAccount OUTPUT;  
SELECT @ServiceAccount;
```

DBA Cheat Sheet

In SQL 2012 and above

```
SELECT
    servicename AS 'Service Name'
    ,service_account
    ,startup_type_desc AS 'Startup Type'
    ,status_desc as 'Status'
    ,last_startup_time as 'Last Startup Time'
```

FROM sys.dm_server_services;

Reference: <http://sqlity.net/en/1868/identify-sql-server-service-account-in-t-sql/>

Find all Available DMVs (SQL 2005 and above)

```
--Find all DMV objects
--Types:
--    V=View
--    TF= SQL table-valued function
--    IF= SQL inlined table-valued function
SELECT * FROM sys.all_objects
WHERE [name] LIKE '%DM_%' AND [type] IN ('V', 'TF', 'IF')
AND [schema_id] = 4; --The sys schema has schema_id =4;
```

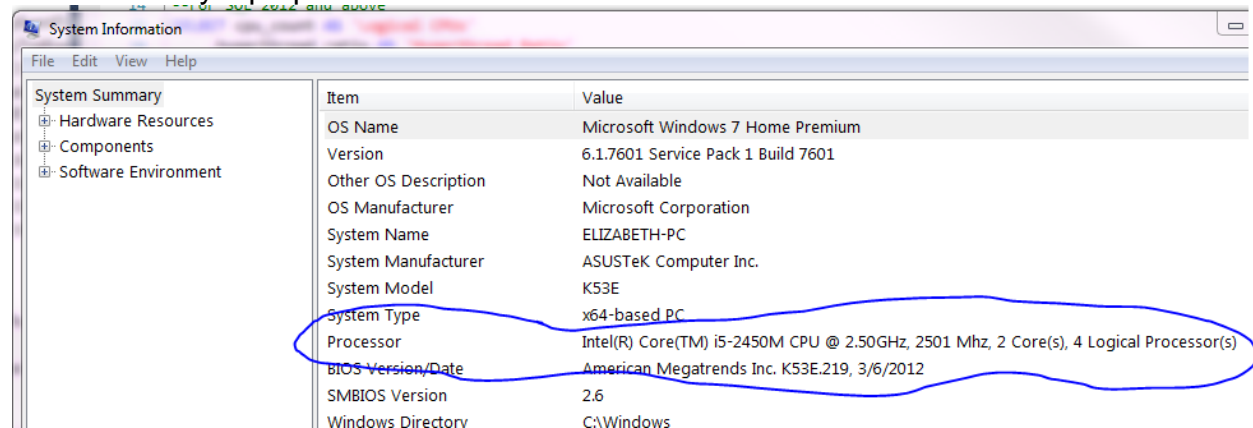
Reference: [https://msdn.microsoft.com/en-us/library/ms188754.aspx#](https://msdn.microsoft.com/en-us/library/ms188754.aspx#Top)

[Top](#)

Hardware Info

Find Number of Processors

To find the number of cores on your server, type msinfo32 in the Windows search box
Results on my laptop:



Item	Value
OS Name	Microsoft Windows 7 Home Premium
Version	6.1.7601 Service Pack 1 Build 7601
Other OS Description	Not Available
OS Manufacturer	Microsoft Corporation
System Name	ELIZABETH-PC
System Manufacturer	ASUSTeK Computer Inc.
System Model	K53E
System Type	x64-based PC
Processor	Intel(R) Core(TM) i5-2450M CPU @ 2.50GHz, 2501 Mhz, 2 Core(s), 4 Logical Processor(s)
BIOS Version/Date	American Megatrends Inc. K53E.219, 3/6/2012
SMBIOS Version	2.6
Windows Directory	C:\Windows

--For SQL 2005/2008

```
SELECT cpu_count AS 'Logical CPUs'
    ,hyperthread_ratio AS 'Hyperthread Ratio'
    ,cpu_count/hyperthread_ratio AS '# of Physical CPU'
    ,physical_memory_in_bytes/1048576 AS 'Physical Memory (MB)'
    --SQL 2008 also has sqlserver_start_time field shown below
FROM sys.dm_os_sys_info;
```

DBA Cheat Sheet

```
--For SQL 2012 and above
SELECT cpu_count AS 'Logical CPUs'
      ,hyperthread_ratio AS 'Hyperthread Ratio'
      ,cpu_count/hyperthread_ratio AS '# of Physical CPU'
      ,physical_memory_kb/1024 AS 'Physical Memory (MB)'
      ,sqlserver_start_time AS 'SQL Server Start Time'
FROM sys.dm_os_sys_info;
Reference: https://msdn.microsoft.com/en-us/library/ms175048.aspx
```

Find Server Memory (SQL 2008 and up)

```
SELECT * FROM sys.dm_os_sys_memory;
Reference: https://msdn.microsoft.com/en-us/library/bb510493\(v=sql.100\).aspx
```

Find Free Drive Space

```
EXEC master..xp_fixeddrives;
Top
```

Database Information

Display Database Compatibility Level

```
SELECT name, compatibility_level
      ,version_name =
      CASE compatibility_level
        WHEN 65 THEN 'SQL Server 6.5'
        WHEN 70 THEN 'SQL Server 7.0'
        WHEN 80 THEN 'SQL Server 2000'
        WHEN 90 THEN 'SQL Server 2005'
        WHEN 100 THEN 'SQL Server 2008/R2'
        WHEN 110 THEN 'SQL Server 2012'
        WHEN 120 THEN 'SQL Server 2014'
        WHEN 130 THEN 'SQL Server 2016'
      END
FROM sys.databases;
```

Change Compatibility Level

```
USE [master]
GO
ALTER DATABASE AdventureWorks SET COMPATIBILITY_LEVEL = 110;
```

List all Databases with dbid , Data Filename

```
SELECT name AS 'Database', dbid, crdate AS 'Create Date', filename as 'Data Filename'
FROM sys.sysdatabases;
```

Find Database ID from the Database Name

```
SELECT db_id(' AdventureWorks ') AS 'Database ID';
```

Find Database Owner for all Databases

SQL 2000 Syntax:

```
SELECT suser_sname(sid) FROM master.dbo.sysdatabases;
```

SQL 2005/2008/2012/2014 Syntax:

DBA Cheat Sheet

```
SELECT name AS 'Database', suser_sname(owner_sid) AS 'Database Owner'
FROM sys.databases;
Reference: https://msdn.microsoft.com/en-us/library/ms178534.aspx
Top
```

Find Database Files, Locations, and File Sizes

```
--List Database Files, locations, and sizes
SELECT
    DB.name AS 'Database',
    MF.Name AS 'Logical File Name',
    MF.physical_name AS 'Physical File',
    MF.state_desc AS 'Status',
    CAST((MF.size*8)/1024 AS VARCHAR(26)) + ' MB' AS 'File Size (MB)',
    CAST(MF.size*8 AS VARCHAR(32)) + ' Bytes' as 'File Size (Bytes)'
FROM
    sys.master_files MF
    INNER JOIN sys.databases DB ON DB.database_id = MF.database_id
ORDER BY
    DB.name;
References: https://msdn.microsoft.com/en-us/library/ms186782.aspx
https://msdn.microsoft.com/en-us/library/ms178534.aspx
```

Find All User Stored Procedures and Functions

System stored procedures are prefixed by “sys”; user stored procedures default to being prefixed by “dbo”. To find both user functions and stored procedures run the following script on the desired database.

```
EXECUTE [dbo].[sp_stored_procedures] @sp_owner = 'dbo';
```

Reference: <https://msdn.microsoft.com/en-us/library/ms190504.aspx>

Find Creation Date of all Stored Procedures and Functions

```
--List of all stored procedures
SELECT * FROM INFORMATION_SCHEMA.ROUTINES
WHERE ROUTINE_TYPE = N'PROCEDURE' and ROUTINE_SCHEMA = N'dbo' ;
--List of all functions
SELECT * FROM INFORMATION_SCHEMA.ROUTINES
WHERE ROUTINE_TYPE = N'FUNCTION' and ROUTINE_SCHEMA = N'dbo' ;
Reference: https://msdn.microsoft.com/en-us/library/ms188757.aspx
Top
```

Find all of the user tables in the current database

```
--Find all the user tables in the current database
SELECT * FROM INFORMATION_SCHEMA.TABLES ORDER BY TABLE_NAME;
```

Reference: <https://msdn.microsoft.com/en-us/library/ms186224.aspx>

Find Last Time Database was Accessed

The code from the website below shows you the last time any index was accessed. This uses a DMV so the information will be cleared out after a server restart.

```
SELECT
last_user_seek = MAX(last_user_seek),
last_user_scan = MAX(last_user_scan),
```

DBA Cheat Sheet

```
last_user_lookup = MAX(last_user_lookup),
last_user_update = MAX(last_user_update)
FROM
sys.dm_db_index_usage_stats
WHERE
```

```
[database_id] = DB_ID();
```

Reference: stackoverflow.com/questions/711394/how-do-you-find-the-last-time-a-database-was-accessed

[Top](#)

Find Database Free Space

```
--Finds the size of the current database
--In first result set:
--      database_size - total size in MB of both data and log files
--      unallocated space - free space in database
--In second result set:
--      data - total space in KB used by data
--      index size - total space in KB used by indexes
EXEC sp_spaceused;
```

Reference: <https://msdn.microsoft.com/en-us/library/ms188776.aspx>

Find Transaction Log Size for All Databases

This command will show the space of the transaction logs for all databases, along with the percentage of the log file that's used.

```
DBCC SQLPERF (LOGSPACE);
```

Reference: <https://msdn.microsoft.com/en-us/library/ms189768.aspx>

[Top](#)

Recovery Model

The recovery model determines whether you can restore to a point in time or not and how long data is kept in the transaction log.

List Recovery Model for all Databases

```
SELECT name AS [Database Name], recovery_model_desc AS [Recovery Model]
FROM sys.databases;
GO
```

References: <https://msdn.microsoft.com/en-us/library/ms189275.aspx>
<https://msdn.microsoft.com/en-us/library/ms189272.aspx>
<https://www.mssqltips.com/sqlservertip/1497/selecting-the-sql-server-database-recovery-model-to-ensure-proper-backups/>

Free eBook: <https://www.red-gate.com/library/sql-server-transaction-log-management>

[Top](#)

Backup

Simple FULL Backup and Verify Backup Commands

To verify that a backup file is good without performing a restore, use RESTORE VERIFY ONLY.

```
BACKUP DATABASE MYDATABASE
TO DISK= 'e:\Microsoft SQL
Server\MSSQL10.MSSQLSERVER\MSSQL\Backup\MYDATABASE_backup_20130919.bak'
WITH STATS = 10;
```

```
RESTORE VERIFYONLY FROM DISK= 'e:\Microsoft SQL
Server\MSSQL10.MSSQLSERVER\MSSQL\Backup\MYDATABASE_backup_20130919.bak';
```

Reference: <https://msdn.microsoft.com/en-us/library/ms186865.aspx>
<https://msdn.microsoft.com/en-us/library/ms188902.aspx>

Create Backup File Name with Date

```
DECLARE @Today varchar(50)
SELECT @Today = REPLACE(REPLACE(REPLACE((CONVERT(nvarchar, GETDATE()), 120)), '-', ''), ' ', '_'), ':', '')
SELECT 'S:\movies_' + @Today + '.bak';
--Results:
--      S:\movies_20150720_152517.bak
```

Backup Database with Date and Time

```
DECLARE @MyFileName varchar(50)
SELECT @MyFileName= 'S:\movies_FULL_' +
REPLACE(REPLACE(REPLACE((CONVERT(nvarchar, GETDATE()), 120)), '-', ''), ' ', '_'), ':', '') +
'.bak'
BACKUP DATABASE movies TO DISK=@MyFileName WITH NOFORMAT, NOINIT, SKIP, STATS=10;
```

Backup Transaction Log

If a database is in Full recovery mode the transaction log needs to be backed up regularly so the log file doesn't grow out of control.

```
--Backup the transaction log to a backup device called Movies_Noon
BACKUP LOG Movies TO Movies_Noon;
```

Reference: <https://msdn.microsoft.com/en-us/library/ms179478.aspx>
<https://msdn.microsoft.com/en-us/library/ms179313.aspx>

Find Databases in Full Recovery without TLog Backups

```
SELECT D.[name] AS [database_name], D.[recovery_model_desc]
FROM sys.databases D LEFT JOIN
(
    SELECT BS.[database_name],
           MAX(BS.[backup_finish_date]) AS [last_log_backup_date]
    FROM msdb.dbo.backupset BS
    WHERE BS.type = 'L'
    GROUP BY BS.[database_name]
) BS1 ON D.[name] = BS1.[database_name]
WHERE D.[recovery_model_desc] <> 'SIMPLE'
AND BS1.[last_log_backup_date] IS NULL
ORDER BY D.[name];
```

Reference: <https://www.mssqltips.com/sqlservertip/1895/script-to-find-sql-server-databases-without-transaction-log-backups/>

[Top](#)

Backup Device

Create Backup Device

Create backup devices for movies database

```
EXEC master.dbo.sp_addumpdevice
    @devtype = N'disk',
    @logicalname = N'movies_0000_Tran',
    @physicalname = N'F:\MSSQL\Backup\Hourly_Backups\movies_0000_Tran.bak';
```

Reference: <https://msdn.microsoft.com/en-us/library/ms188409%28v=sql.105%29.aspx>

List All Backup Devices

```
SELECT * FROM sys.backup_devices;
```

Delete Backup Device

```
EXEC sp_dropdevice 'movies_0600__MWF_Diff';
```

Reference: <https://msdn.microsoft.com/en-us/library/ms188903.aspx>

[Top](#)

Restore

Simple Restore

This does a simple full database restore of the MySampleDB database.

First it finds the logical filenames using the query below. This is needed if the location of the data files is different from the source system, which is common in a refresh from production to test. In our organization often the test database server only has 2 drives while production has at least 3 drives.

```
sp_helpdb MySampleDB;
```

Results:

name	filename
	E:\Microsoft SQL
MySampleDB1_Data	Server\MSSQL10.MSSQLSERVER\MSSQL\DATA\MySampleDB.MDF
	E:\Microsoft SQL
MySampleDB1_Log	Server\MSSQL10.MSSQLSERVER\MSSQL\DATA\MySampleDB_log.ldf

If the logical file name doesn't match the data file name or the location is not the default database location you'll need to include the move arguments.

```
--Verify no one is connected to the database to restore. You can't restore a database that anyone is connected to.
```

```
sp_who2;
```

```
RESTORE DATABASE MySampleDB FROM DISK= 'E:\Restore\MySampleDB_FULL_20150727_232123.bak'
WITH FILE = 1,
MOVE N'MySampleDB1_Data' TO N'E:\Microsoft SQL
Server\MSSQL10.MSSQLSERVER\MSSQL\DATA\MySampleDB.MDF',
```

DBA Cheat Sheet

```
MOVE N'MySampleDB1_Log' TO N'E:\Microsoft SQL
Server\MSSQL10.MSSQLSERVER\MSSQL\DATA\MySampleDB_log.ldf',
REPLACE, STATS = 10;
```

Restore Database to Different Location than Backup Source

If the dev/test server has different drive configuration than the production system, when you restore a production database to a dev/test server you'll need to use the MOVE option.

```
RESTORE DATABASE Movies
FROM DISK= 'E:\Restore\Movies_FULL_20131217_232618.bak'
WITH RECOVERY,REPLACE,STATS = 10,
MOVE 'Movies' TO 'E:\Microsoft SQL Server\MSSQL10.MSSQLSERVER\MSSQL\DATA\Movies.mdf',
MOVE 'Movies_log' TO 'E:\Microsoft SQL
Server\MSSQL10.MSSQLSERVER\MSSQL\DATA\Movies_log.ldf';
```

Find Last Restore Date of Database

This script lists the last restore date of the desired database using the history tables in the msdb database.

```
DECLARE @dbname sysname, @days int
SET @dbname = 'TSQL2012' --substitute for whatever database name you want
SET @days = -30 --previous number of days, script will default to 30
SELECT
  rsh.destination_database_name AS [Database],
  rsh.user_name AS [Restored By],
  CASE WHEN rsh.restore_type = 'D' THEN 'Database'
  WHEN rsh.restore_type = 'F' THEN 'File'
  WHEN rsh.restore_type = 'G' THEN 'Filegroup'
  WHEN rsh.restore_type = 'I' THEN 'Differential'
  WHEN rsh.restore_type = 'L' THEN 'Log'
  WHEN rsh.restore_type = 'V' THEN 'Verifyonly'
  WHEN rsh.restore_type = 'R' THEN 'Revert'
  ELSE rsh.restore_type
  END AS [Restore Type],
  rsh.restore_date AS [Restore Started],
  bmf.physical_device_name AS [Restored From],
  rf.destination_phys_name AS [Restored To]
FROM msdb.dbo.restorehistory rsh
  INNER JOIN msdb.dbo.backupset bs ON rsh.backup_set_id = bs.backup_set_id
  INNER JOIN msdb.dbo.restorefile rf ON rsh.restore_history_id = rf.restore_history_id
  INNER JOIN msdb.dbo.backupmediafamily bmf ON bmf.media_set_id = bs.media_set_id
WHERE rsh.restore_date >= DATEADD(dd, ISNULL(@days, -30), GETDATE()) --want to search for
previous days
AND destination_database_name = ISNULL(@dbname, destination_database_name) --if no
dbname, then return all
ORDER BY rsh.restore_history_id DESC;
GO
```

Reference: <http://www.mssqltips.com/sqlservertip/1724/when-was-the-last-time-your-sql-server-database-was-restored/>

[Top](#)

SQL Agent Jobs

List Jobs and Job Steps

```
--List all jobs that contain "TTHSA" in their name"
USE msdb
SELECT * FROM sysjobs WHERE name LIKE '%TTHSA%'
ORDER BY name;
--List all job steps that have "%TTHSA " their step_name
SELECT * FROM sysjobsteps WHERE step_name LIKE '%TTHSA %'
ORDER BY step_name;
```

Create Job

Steps to create a SQL Agent job:

1. Use sp_add_job to create the job
2. Use sp_add_jobstep to create the job steps
3. Use sp_add_schedule to specify when the job will run
4. Use sp_attache_schedule to connect the job schedule to the job
5. Use sp_add_jobserver to specify the server that will run the job

Reference: <https://technet.microsoft.com/en-us/library/ms190268.aspx#TsqlProcedure>
[https://technet.microsoft.com/en-us/library/ms181153\(v=sql.105\).aspx](https://technet.microsoft.com/en-us/library/ms181153(v=sql.105).aspx)

Delete Job

```
USE msdb;
GO
EXEC sp_delete_job
    @job_name = N'NightlyBackups';
GO
```

Disabling Jobs

```
USE msdb
GO
--Use this script to find the name of the job to disable
SELECT * FROM sysjobs ORDER BY name;
```

Disable one job

```
EXEC msdb.dbo.sp_update_job
    @job_name='My job name',
    @enabled = 0 ;
```

Enable a job - set @enabled = 1

Reference: <https://msdn.microsoft.com/en-us/library/ms186976.aspx>

Disable all the enabled jobs containing "Hourly_Backup" in their names

```
-- Creates the statements to disable specific SQL Agent jobs
DECLARE @sql NVARCHAR(max) = ''
SELECT
    @sql += N'exec msdb.dbo.sp_update_job @job_name = ''' + name + N''', @enabled = 0;
' FROM msdb.dbo.sysjobs
WHERE enabled = 1 and name LIKE '%Hourly_Backup%'
ORDER BY name;

PRINT @sql;
--EXEC (@sql);
GO
```

DBA Cheat Sheet

Change the owner of a job to sa

```
DECLARE @sql NVARCHAR(max) = ''
SELECT
    @sql += N'exec msdb.dbo.sp_update_job @job_name = '' + name + N'',
    @owner_login_name = ''sa'';
' FROM msdb.dbo.sysjobs
WHERE enabled = 1 AND name LIKE '%CDW%'
ORDER BY name;
PRINT @sql;
--EXEC (@sql);
GO
```

Reference: <http://davidbrycehoward.com/archive/2011/03/disable-all-sql-agent-jobs/>
[Top](#)

Training Info

Free Newsletters

- <https://www.brentozar.com>
- <https://www.mssqltips.com/sql-server-tutorials/>
- <http://www.sqlservercentral.com/>

Free eBooks

- http://www.red-gate.com/community/books/?&qclid=CjwKEAjw0KK4BRDCiKHD5Ny8pHESJACLE620CHtFEUFOP6DBUVUZI7tZLF873bUSgp4domo4GsOXSRoC9HHw_wcB
- <https://www.brentozar.com/first-aid/free-database-books-pdfs-download/>
- <https://www.toadworld.com/platforms/sql-server/b/weblog/archive/2013/06/21/huge-collection-of-free-microsoft-sql-server-ebooks>

SSMS Tips

- Keyboard Shortcuts: <https://msdn.microsoft.com/en-us/library/ms174205.aspx?f=255&MSPPErr=-2147217396>
- Registered Servers: <https://www.mssqltips.com/sqlservertip/1996/registering-sql-server-instances-for-easier-management/>
- Import and Export Registered Servers: <https://www.mssqltips.com/sqlservertip/2015/import-and-export-registered-sql-servers-to-other-machines/>
- Display Line Numbers in Query Window: <https://www.mssqltips.com/sqlservertip/2542/display-line-numbers-in-a-sql-server-management-studio-query-window/>
- Assign Colors in SSMS Based on Environment: <https://www.mssqltips.com/sqlservertip/3565/assign-colors-in-sql-server-management-studio-query-windows-based-on-environment/>
- Getting Started with Code Snippets: <https://www.mssqltips.com/sqlservertip/2411/getting-started-with-code-snippets-feature-of-sql-server-2012/>
- Cycle Clipboard Ring <https://www.mssqltips.com/sqlservertip/2449/sql-server-2012-cycle-clipboard-ring-feature/>
- Zoom Feature for Query Text: <https://www.mssqltips.com/sqlservertip/2374/zoom-feature-for-query-text-and-results-text-in-sql-server-management-studio/>

[Top](#)

Websites

PASS

- 24 Hours of PASS: <http://www.sqlpass.org/Events/24HoursofPASS.aspx>
- SQL Saturday: <http://www.sqlsaturday.com/>
- Virtual Chapters: <http://www.sqlpass.org/PASSChapters/VirtualChapters.aspx>
- White papers, session recordings: <http://www.sqlpass.org/LearningCenter.aspx>

Brent Ozar

- Training Videos: <https://learnfrom.brentozar.com/>
- First Aid Scripts: <https://www.brentozar.com/first-aid/>

MSSQL Tips: <https://www.mssqltips.com/sql-server-dba-resources/>

Pragmatic Works

- Free Webinars: <http://pragmaticworks.com/Training/Courses#subject=SQL-Server>

SQL Server Central

- Stairway Articles: <http://www.sqlservercentral.com/stairway/>
- Forums: <http://www.sqlservercentral.com/Forums/Forum3411-1.aspx>

SQL Skills: <https://www.sqlskills.com/sql-server-resources/>

- Whitepapers: <https://www.sqlskills.com/sql-server-resources/sql-server-whitepapers/>
- Resources: <https://www.sqlskills.com/sql-server-resources/>

[Top](#)

Free Training Links

Brent Ozar

- <https://www.brentozar.com/sql-server-training-videos/>

Pragmatic Works

- Pragmatic Works gives free webinars every Tuesday:
<http://pragmaticworks.com/Training/Courses#type=Free>

SQL Skills

- These videos are geared towards the material needed to pass the MCM 2008 exam. While the information is a little dated, a lot of it is still very helpful.
<https://www.sqlskills.com/sql-server-resources/sql-server-mcm-training-videos/>

Online Training

- Brent Ozar: <https://learnfrom.brentozar.com/>
- Pluralsight: <https://www.pluralsight.com/pricing>
- Pragmatic Works: <http://pragmaticworks.com/Training/Courses#type=On-Demand>
- SQL Skills: <https://www.sqlskills.com/sql-server-training/>
- CBT Nuggets: <https://www.cbtnuggets.com>

[Top](#)