MsSQL - Performance Script and Monitoring Scripts

High usage

```
SELECT TOP 10
    qs.total_worker_time / 1000 AS [Total CPU Time (ms)],
    qs.execution_count AS [Execution Count],
    qs.total_worker_time / qs.execution_count / 1000 AS [Avg CPU Time (ms)],
    qs.total_elapsed_time / qs.execution_count / 1000 AS [Avg Elapsed Time (ms)],
    qs.total_logical_reads / qs.execution_count AS [Avg Logical Reads],
    SUBSTRING(st.text, (qs.statement_start_offset/2) + 1,
        ((CASE qs.statement_end_offset
            WHEN -1 THEN DATALENGTH(st.text)
            ELSE qs.statement_end_offset
        END - qs.statement_start_offset)/2) + 1) AS [Query Text],
    qp.query_plan AS [Query Plan]
FROM sys.dm_exec_query_stats qs
CROSS APPLY sys.dm_exec_sql_text(qs.sql_handle) st
CROSS APPLY sys.dm_exec_query_plan(qs.plan_handle) qp
ORDER BY qs.total_worker_time DESC;
```

SQL Messages

```
SELECT sql_message_id, message, run_date, run_time
FROM msdb.dbo.sysjobhistory h
JOIN msdb.dbo.sysjobsteps s ON h.job_id = s.job_id AND h.step_id = s.step_id
WHERE h.job_id = (SELECT job_id FROM msdb.dbo.sysjobs WHERE name = '_MAINT_DatabaseBackup -
USER_DATABASES - FULL')
ORDER BY run_date DESC, run_time DESC;
```

Note: sys.dm_exec_query_stats reports worker and elapsed time in microseconds, so the query divides by 1000 to show milliseconds.

Last verified: October 2026

Sources:

- https://learn.microsoft.com/en-us/sql/relational-databases/system-dynamic-management-views/sys-dm-exec-query-stats-transact-sql

---

Revision #1
Created 6 August 2026 23:11:07 by Admin
Updated 8 October 2026 15:26:51 by Admin