USE [Admin]

GO

/\*\*\*\*\*\* Object: StoredProcedure [dbo].[sp\_loadpro] Script Date: 11/19/2023 9:44:35 AM \*\*\*\*\*\*/

SET ANSI\_NULLS ON

GO

SET QUOTED\_IDENTIFIER ON

GO

CREATE PROCEDURE [dbo].[sp\_loadpro]

@start\_time DATETIME = NULL,

@end\_time DATETIME = NULL,

@top\_rows INT = NULL

AS

BEGIN

SET NOCOUNT ON;

-- the logic for analyzing CPU load, using the whoisactive table

SELECT TOP (ISNULL(@top\_rows, 100))

[CPU],

[reads],

[collection\_time],

[start\_time],

[dd hh:mm:ss.mss] AS 'run duration',

[program\_name],

[login\_name],

[database\_name],

[session\_id],

[blocking\_session\_id],

[wait\_info],

[sql\_text]

FROM whoisactive

WHERE [collection\_time] BETWEEN @start\_time AND @end\_time OR (@start\_time IS NULL AND @end\_time IS NULL)

ORDER BY [CPU] DESC, [collection\_time] DESC;

END;

GO

USE [Admin]

GO

/\*\*\*\*\*\* Object: StoredProcedure [dbo].[sp\_blockpro] Script Date: 11/19/2023 9:43:39 AM \*\*\*\*\*\*/

SET ANSI\_NULLS ON

GO

SET QUOTED\_IDENTIFIER ON

GO

CREATE PROCEDURE [dbo].[sp\_blockpro]

@start\_time DATETIME = NULL,

@end\_time DATETIME = NULL,

@top\_rows INT = NULL

AS

BEGIN

SET NOCOUNT ON;

-- the logic for analyzing blocking, using the whoisactive table

SELECT TOP (CASE WHEN @top\_rows IS NOT NULL THEN @top\_rows ELSE 100 END)

[blocking\_session\_id] AS 'Blocker spid',

[session\_id] AS 'Blocked spid',

[reads],

[writes],

[collection\_time],

[start\_time],

[dd hh:mm:ss.mss] AS 'Run Duration',

[program\_name],

[login\_name],

[database\_name],

[wait\_info],

[sql\_text]

FROM whoisactive

WHERE [blocking\_session\_id] IS NOT NULL

AND ([collection\_time] BETWEEN @start\_time AND @end\_time OR (@start\_time IS NULL AND @end\_time IS NULL))

ORDER BY [blocking\_session\_id] DESC, [collection\_time] DESC;

END;

GO

USE [Admin]

GO

/\*\*\*\*\*\* Object: StoredProcedure [dbo].[sp\_tempdbpro] Script Date: 11/19/2023 9:44:52 AM \*\*\*\*\*\*/

SET ANSI\_NULLS ON

GO

SET QUOTED\_IDENTIFIER ON

GO

CREATE PROCEDURE [dbo].[sp\_tempdbpro]

@start\_time DATETIME = NULL,

@end\_time DATETIME = NULL,

@top\_rows INT = NULL

AS

BEGIN

SET NOCOUNT ON;

-- the logic for analyzing tempdb activity, using the whoisactive table

SELECT TOP (CASE WHEN @top\_rows IS NOT NULL THEN @top\_rows ELSE 100 END)

[session\_id],

[tempdb\_allocations] AS 'TempDB Allocations',

[tempdb\_current],

[collection\_time],

[start\_time],

[reads] ,

[writes] ,

[dd hh:mm:ss.mss] AS 'Run Duration',

[program\_name],

[login\_name],

[database\_name],

[sql\_text]

FROM whoisactive

WHERE [collection\_time] BETWEEN @start\_time AND @end\_time OR (@start\_time IS NULL AND @end\_time IS NULL)

ORDER BY [tempdb\_allocations] DESC, [collection\_time] DESC;

END;

GO

**LOADPRO: ANALYSE CPU LOAD**

-- Execute sp\_loadpro for the last hour and show the top rows (default 100)

EXEC dbo.sp\_loadpro '2023-11-17 00:00:00', '2023-11-17 19:00:00';

-- Execute sp\_loadpro for the last hour and show the top 50 rows

EXEC dbo.sp\_loadpro '2023-11-17 00:00:00', '2023-11-17 19:00:00', 50;

-- Execute sp\_loadpro without specifying a time interval and show the top rows (default 100)

EXEC dbo.sp\_loadpro;

**BLOCKPRO: ANALYSE BLOCKING**

-- Execute sp\_blockpro for the last hour and show the top rows

EXEC dbo.sp\_blockpro '2023-11-17 09:52:00', '2023-11-17 10:58:00';

-- Execute sp\_blockpro for the last hour and show the top 50 rows

EXEC dbo.sp\_blockpro '2023-11-17 00:00:00', '2023-11-17 19:00:00', 50;

-- Execute sp\_blockpro without specifying a time interval and show the top rows

EXEC dbo.sp\_blockpro;

**TEMPDBPRO: TEMPDB USAGE**

-- Execute sp\_tempdbpro for the last hour and show the top rows

EXEC dbo.sp\_tempdbpro '2023-01-01 00:00:00', '2023-01-02 00:00:00';

-- Execute sp\_tempdbpro for the last hour and show the top 50 rows

EXEC dbo.sp\_tempdbpro '2023-01-01 00:00:00', '2023-01-02 00:00:00', 50;

-- Execute sp\_tempdbpro without specifying a time interval and show the top rows

EXEC dbo.sp\_tempdbpro;