/\*\*\*\*\*\* Object: StoredProcedure [dbo].[SP\_GET\_TASK\_RECURSIVE] Script Date: 11-10-2024 16:18:21 \*\*\*\*\*\*/

SET ANSI\_NULLS ON

GO

SET QUOTED\_IDENTIFIER ON

GO

-- =============================================

-- AUTHOR: AMIT SHUDDHALWAR

-- CREATE DATE: 19-09-2024

-- DESCRIPTION: TO GET TASK RECURSIVE TABLE DETAILS (HDR,TRL)

-- EXEC SP\_GET\_TASK\_RECURSIVE @MKEY=NULL

-- EXEC SP\_GET\_TASK\_RECURSIVE @MKEY=102,3

-- =============================================

ALTER PROCEDURE [dbo].[SP\_GET\_TASK\_RECURSIVE]

-- ADD THE PARAMETERS FOR THE STORED PROCEDURE HERE

@MKEY NUMERIC = NULL

AS

BEGIN

-- SET NOCOUNT ON ADDED TO PREVENT EXTRA RESULT SETS FROM

-- INTERFERING WITH SELECT STATEMENTS.

SET NOCOUNT ON;

BEGIN TRY ;

-- TO CHECK THE @MKEY IS NULL OR EMPTY IF EMPTY THE SHOW ALL ROWS

IF ( @MKEY IS NOT NULL

AND LEN(@MKEY) > 0 )

BEGIN

SELECT HDR.MKEY,

HDR.TASK\_NAME,

HDR.TASK\_DESCRIPTION,

HDR.TERM,

CONVERT(DATETIME2,HDR.[START\_DATE]) AS 'START\_DATE',

HDR.ENDS,

CONVERT(DATETIME2,HDR.END\_DATE) AS 'END\_DATE',

HDR.STATUS,

HDR.CREATED\_BY,

HDR.CREATION\_DATE,

HDR.LAST\_UPDATED\_BY,

HDR.LAST\_UPDATE\_DATE,

TRL.ATTRIBUTE1,

TRL.ATTRIBUTE2,

TRL.ATTRIBUTE3,

TRL.ATTRIBUTE4,

TRL.ATTRIBUTE5,

TRL.ATTRIBUTE6,

TRL.ATTRIBUTE7,

TRL.ATTRIBUTE8,

TRL.ATTRIBUTE9,

TRL.ATTRIBUTE10,

TRL.ATTRIBUTE11,

TRL.ATTRIBUTE12,

TRL.ATTRIBUTE13,

TRL.SR\_NO,

TRL.TERM\_TYPE,

TRL.CREATED\_BY,

TRL.CREATION\_DATE,

TRL.LAST\_UPDATED\_BY,

TRL.LAST\_UPDATE\_DATE

FROM [DBO].[TASK\_RECURSIVE\_HDR] HDR

INNER JOIN [DBO].[TASK\_RECURSIVE\_TRL] TRL

ON HDR.MKEY = TRL.MKEY

WHERE HDR.MKEY in(@MKEY)

AND HDR.DELETE\_FLAG = 'N'

AND TRL.DELETE\_FLAG = 'N'

ORDER BY HDR.MKEY;

END

ELSE

BEGIN

SELECT HDR.MKEY,

HDR.TASK\_NAME,

HDR.TASK\_DESCRIPTION,

HDR.TERM,

CONVERT(DATETIME2,HDR.[START\_DATE]) AS 'START\_DATE',

HDR.ENDS,

CONVERT(DATETIME2,HDR.END\_DATE) AS 'END\_DATE',

HDR.STATUS,

HDR.CREATED\_BY,

HDR.CREATION\_DATE,

HDR.LAST\_UPDATED\_BY,

HDR.LAST\_UPDATE\_DATE,

HDR.DELETE\_FLAG,

HDR.IS\_SCHEDULED,

TRL.ATTRIBUTE1,

TRL.ATTRIBUTE2,

TRL.ATTRIBUTE3,

TRL.ATTRIBUTE4,

TRL.ATTRIBUTE5,

TRL.ATTRIBUTE6,

TRL.ATTRIBUTE7,

TRL.ATTRIBUTE8,

TRL.ATTRIBUTE9,

TRL.ATTRIBUTE10,

TRL.ATTRIBUTE11,

TRL.ATTRIBUTE12,

TRL.ATTRIBUTE13,

TRL.MKEY,

TRL.SR\_NO,

TRL.TERM\_TYPE,

TRL.CREATED\_BY,

TRL.CREATION\_DATE,

TRL.LAST\_UPDATED\_BY,

TRL.LAST\_UPDATE\_DATE,

TRL.DELETE\_FLAG

FROM [DBO].[TASK\_RECURSIVE\_HDR] HDR

INNER JOIN [DBO].[TASK\_RECURSIVE\_TRL] TRL

ON HDR.MKEY = TRL.MKEY

WHERE HDR.DELETE\_FLAG = 'N'

ORDER BY HDR.MKEY;

END

END TRY

BEGIN CATCH

SELECT ERROR\_NUMBER() AS ERRORNUMBER,

ERROR\_SEVERITY() AS ERRORSEVERITY,

ERROR\_STATE() AS ERRORSTATE,

ERROR\_PROCEDURE() AS ERRORPROCEDURE,

ERROR\_LINE() AS ERRORLINE,

ERROR\_MESSAGE() AS ERRORMESSAGE;

END CATCH

END

------------------------------------------------------------------------------------------------------------------------------------------

/\*\*\*\*\*\* Object: StoredProcedure [dbo].[SP\_INSERT\_TASK\_RECURSIVE\_DETAILS] Script Date: 11-10-2024 16:19:32 \*\*\*\*\*\*/

SET ANSI\_NULLS ON

GO

SET QUOTED\_IDENTIFIER ON

GO

-- =============================================

-- AUTHOR: AMIT SHUDDHALWAR

-- CREATE DATE: 19-09-2024

-- DESCRIPTION: TO INTSERT TASK RECURSIVE HDR AND TRL DETAILS

-- =============================================

ALTER PROCEDURE [dbo].[SP\_INSERT\_TASK\_RECURSIVE\_DETAILS]

@TASK\_NAME VARCHAR(250) = NULL,

@TASK\_DESCRIPTION VARCHAR(250) = NULL,

@TERM VARCHAR(250) = NULL,

@START\_DATE DATETIME,

@ENDS VARCHAR(250) = NULL,

@END\_DATE DATETIME = NULL,

@CREATED\_BY NUMERIC,

@LAST\_UPDATED\_BY NUMERIC,

@ATTRIBUTE1 VARCHAR(250) = NULL,

@ATTRIBUTE2 VARCHAR(250) = NULL,

@ATTRIBUTE3 VARCHAR(250) = NULL,

@ATTRIBUTE4 VARCHAR(250) = NULL,

@ATTRIBUTE5 VARCHAR(250) = NULL,

@ATTRIBUTE6 VARCHAR(250) = NULL,

@ATTRIBUTE7 VARCHAR(250) = NULL,

@ATTRIBUTE8 VARCHAR(250) = NULL,

@ATTRIBUTE9 VARCHAR(250) = NULL,

@ATTRIBUTE10 VARCHAR(250) = NULL,

@ATTRIBUTE11 VARCHAR(250) = NULL,

@ATTRIBUTE12 VARCHAR(250) = NULL,

@ATTRIBUTE13 VARCHAR(250) = NULL

AS

BEGIN

-- SET NOCOUNT ON ADDED TO PREVENT EXTRA RESULT SETS FROM

-- INTERFERING WITH SELECT STATEMENTS.

SET NOCOUNT ON;

DECLARE @SR\_NO NUMERIC, @MKEY NUMERIC;

BEGIN TRY

-- INSERT STATEMENTS FOR PROCEDURE HERE

INSERT INTO TASK\_RECURSIVE\_HDR

(TASK\_NAME,

TASK\_DESCRIPTION,

TERM,

[START\_DATE],

ENDS,

CREATED\_BY,

CREATION\_DATE,

LAST\_UPDATED\_BY,

LAST\_UPDATE\_DATE)

VALUES (@TASK\_NAME,

@TASK\_DESCRIPTION,

@TERM,

@START\_DATE,

@ENDS,

@CREATED\_BY,

GETDATE(),

@LAST\_UPDATED\_BY,

GETDATE());

-- TO GET LATEST MKEY VALUE FROM TASK\_RECURSIVE\_HDR TABLE

SET @MKEY = SCOPE\_IDENTITY();

SELECT @SR\_NO = ISNULL(MAX(SR\_NO),0)+1 FROM TASK\_RECURSIVE\_TRL WHERE MKEY = @MKEY;

INSERT INTO TASK\_RECURSIVE\_TRL

(MKEY,

SR\_NO,

TERM\_TYPE,

CREATED\_BY,

CREATION\_DATE,

LAST\_UPDATED\_BY,

LAST\_UPDATE\_DATE,

ATTRIBUTE1,

ATTRIBUTE2,

ATTRIBUTE3,

ATTRIBUTE4,

ATTRIBUTE5,

ATTRIBUTE6,

ATTRIBUTE7,

ATTRIBUTE8,

ATTRIBUTE9,

ATTRIBUTE10,

ATTRIBUTE11,

ATTRIBUTE12,

ATTRIBUTE13)

VALUES (@MKEY,

@SR\_NO,

@TERM,

@CREATED\_BY,

GETDATE(),

@LAST\_UPDATED\_BY,

GETDATE(),

@ATTRIBUTE1,

@ATTRIBUTE2,

@ATTRIBUTE3,

@ATTRIBUTE4,

@ATTRIBUTE5,

@ATTRIBUTE6,

@ATTRIBUTE7,

@ATTRIBUTE8,

@ATTRIBUTE9,

@ATTRIBUTE10,

@ATTRIBUTE11,

@ATTRIBUTE12,

@ATTRIBUTE13);

-- IF THE END DATE IS NOT THEN IT WILL UPDATE

IF(@END\_DATE IS NOT NULL OR @END\_DATE != NULL)

BEGIN

UPDATE TASK\_RECURSIVE\_HDR

SET END\_DATE = @END\_DATE

WHERE MKEY = @MKEY;

END

SELECT @MKEY;

END TRY

BEGIN CATCH

SELECT ERROR\_NUMBER() AS ERRORNUMBER,

ERROR\_SEVERITY() AS ERRORSEVERITY,

ERROR\_STATE() AS ERRORSTATE,

ERROR\_PROCEDURE() AS ERRORPROCEDURE,

ERROR\_LINE() AS ERRORLINE,

ERROR\_MESSAGE() AS ERRORMESSAGE;

END CATCH

END

------------------------------------------------------------------------------------------------------------------------------------------

/\*\*\*\*\*\* Object: StoredProcedure [dbo].[SP\_UPDATE\_TASK\_RECURSIVE\_DETAILS] Script Date: 11-10-2024 16:20:09 \*\*\*\*\*\*/

SET ANSI\_NULLS ON

GO

SET QUOTED\_IDENTIFIER ON

GO

-- =============================================

-- AUTHOR: AMIT SHUDDHALWAR

-- CREATE DATE: 19-09-2024

-- DESCRIPTION: UPDATE THE DETAILS OF TASK RECURSIVE DETAILS

-- =============================================

ALTER PROCEDURE [dbo].[SP\_UPDATE\_TASK\_RECURSIVE\_DETAILS]

-- ADD THE PARAMETERS FOR THE STORED PROCEDURE HERE

@MKEY INT,

@TASK\_NAME VARCHAR(250) = NULL,

@TASK\_DESCRIPTION VARCHAR(250) = NULL,

@TERM VARCHAR(250) = NULL,

@START\_DATE DATETIME = NULL,

@ENDS VARCHAR(250) = NULL,

@END\_DATE DATETIME = NULL,

@CREATED\_BY NUMERIC = NULL,

@LAST\_UPDATED\_BY NUMERIC = NULL,

@ATTRIBUTE1 VARCHAR(250) = NULL,

@ATTRIBUTE2 VARCHAR(250) = NULL,

@ATTRIBUTE3 VARCHAR(250) = NULL,

@ATTRIBUTE4 VARCHAR(250) = NULL,

@ATTRIBUTE5 VARCHAR(250) = NULL,

@ATTRIBUTE6 VARCHAR(250) = NULL,

@ATTRIBUTE7 VARCHAR(250) = NULL,

@ATTRIBUTE8 VARCHAR(100) = NULL,

@ATTRIBUTE9 VARCHAR(100) = NULL,

@ATTRIBUTE10 VARCHAR(100) = NULL,

@ATTRIBUTE11 VARCHAR(100) = NULL,

@ATTRIBUTE12 VARCHAR(100) = NULL,

@ATTRIBUTE13 VARCHAR(100) = NULL

AS

BEGIN

-- SET NOCOUNT ON ADDED TO PREVENT EXTRA RESULT SETS FROM

-- INTERFERING WITH SELECT STATEMENTS.

SET NOCOUNT ON;

--BEGIN TRY

-- TO CHECK THE MKEY IS PRESENT OR NOT IN THE TASK\_RECURSIVE\_HDR TABLE

--IF ((SELECT COUNT(\*) FROM TASK\_RECURSIVE\_HDR WHERE MKEY = @MKEY AND DELETE\_FLAG = 'N') > 0)

BEGIN

---------------------------------------------------------------------------------------------------------------------

-- TO ADD THE ROW IN HISTORY TABLE FROM MAIN TABLE

INSERT INTO TASK\_RECURSIVE\_HDR\_H

(TASK\_NAME,

TASK\_DESCRIPTION,

TERM,

[START\_DATE],

ENDS,

END\_DATE,

CREATED\_BY,

CREATION\_DATE,

LAST\_UPDATED\_BY,

LAST\_UPDATE\_DATE)

SELECT TASK\_NAME,

TASK\_DESCRIPTION,

TERM,

[START\_DATE],

ENDS,

END\_DATE,

CREATED\_BY,

CREATION\_DATE,

LAST\_UPDATED\_BY,

LAST\_UPDATE\_DATE

FROM TASK\_RECURSIVE\_HDR

WHERE MKEY = @MKEY;

---------------------------------------------------------------------------------------------------------------------

-- TO UPDATE THE ROWS IN MAIN TABLE

-- IF ANY COLUMN ROW VALUE IS NULL THEN UPDATE OLD VALUE

UPDATE TASK\_RECURSIVE\_HDR

SET TASK\_NAME = ISNULL(@TASK\_NAME, TASK\_NAME),

TASK\_DESCRIPTION = ISNULL(@TASK\_DESCRIPTION, TASK\_DESCRIPTION),

TERM = ISNULL(@TERM, TERM),

[START\_DATE] = ISNULL(@START\_DATE, [START\_DATE]),

ENDS = ISNULL(@ENDS, ENDS),

END\_DATE = ISNULL(@END\_DATE, @END\_DATE),

LAST\_UPDATED\_BY = @LAST\_UPDATED\_BY,

LAST\_UPDATE\_DATE = GETDATE()

WHERE MKEY = @MKEY

AND DELETE\_FLAG = 'N';

---------------------------------------------------------------------------------------------------------------------

-- TO INSERT TRL TASK RECURSIVE HISTORY TABLE

INSERT INTO TASK\_RECURSIVE\_TRL\_H

(MKEY,

TERM\_TYPE,

CREATED\_BY,

CREATION\_DATE,

LAST\_UPDATED\_BY,

LAST\_UPDATE\_DATE,

ATTRIBUTE1,

ATTRIBUTE2,

ATTRIBUTE3,

ATTRIBUTE4,

ATTRIBUTE5,

ATTRIBUTE6,

ATTRIBUTE7,

ATTRIBUTE8,

ATTRIBUTE9,

ATTRIBUTE10,

ATTRIBUTE11,

ATTRIBUTE12,

ATTRIBUTE13)

SELECT MKEY,

TERM\_TYPE,

CREATED\_BY,

CREATION\_DATE,

LAST\_UPDATED\_BY,

LAST\_UPDATE\_DATE,

ATTRIBUTE1,

ATTRIBUTE2,

ATTRIBUTE3,

ATTRIBUTE4,

ATTRIBUTE5,

ATTRIBUTE6,

ATTRIBUTE7,

ATTRIBUTE8,

ATTRIBUTE9,

ATTRIBUTE10,

ATTRIBUTE11,

ATTRIBUTE12,

ATTRIBUTE13

FROM TASK\_RECURSIVE\_TRL

WHERE MKEY = @MKEY;

---------------------------------------------------------------------------------------------------------------------

-- TO UPDATE THE MAIN TRL TABLE

UPDATE TASK\_RECURSIVE\_TRL

SET TERM\_TYPE = ISNULL(@TERM, TERM\_TYPE),

LAST\_UPDATED\_BY = @LAST\_UPDATED\_BY,

LAST\_UPDATE\_DATE = GETDATE(),

ATTRIBUTE1 = ISNULL(@ATTRIBUTE1, ATTRIBUTE1),

ATTRIBUTE2 = ISNULL(@ATTRIBUTE2, ATTRIBUTE2),

ATTRIBUTE3 = ISNULL(@ATTRIBUTE3, ATTRIBUTE3),

ATTRIBUTE4 = ISNULL(@ATTRIBUTE4, ATTRIBUTE4),

ATTRIBUTE5 = ISNULL(@ATTRIBUTE5, ATTRIBUTE5),

ATTRIBUTE6 = ISNULL(@ATTRIBUTE6, ATTRIBUTE6),

ATTRIBUTE7 = ISNULL(@ATTRIBUTE7, ATTRIBUTE7),

ATTRIBUTE8 = ISNULL(@ATTRIBUTE8, ATTRIBUTE8),

ATTRIBUTE9 = ISNULL(@ATTRIBUTE9, ATTRIBUTE9),

ATTRIBUTE10 = ISNULL(@ATTRIBUTE10, ATTRIBUTE10),

ATTRIBUTE11 = ISNULL(@ATTRIBUTE11, ATTRIBUTE11),

ATTRIBUTE12 = ISNULL(@ATTRIBUTE12, ATTRIBUTE12),

ATTRIBUTE13 = ISNULL(@ATTRIBUTE13, ATTRIBUTE13)

WHERE MKEY = @MKEY

AND DELETE\_FLAG = 'N';

---------------------------------------------------------------------------------------------------------------------

END

--END TRY

-- BEGIN CATCH

-- SELECT

-- ERROR\_NUMBER() AS ERRORNUMBER

-- ,ERROR\_SEVERITY() AS ERRORSEVERITY

-- ,ERROR\_STATE() AS ERRORSTATE

-- ,ERROR\_PROCEDURE() AS ERRORPROCEDURE

-- ,ERROR\_LINE() AS ERRORLINE

-- ,ERROR\_MESSAGE() AS ERRORMESSAGE;

-- END CATCH

END

------------------------------------------------------------------------------------------------------------------------------------------

GO

/\*\*\*\*\*\* Object: StoredProcedure [dbo].[SP\_DELETE\_TASK\_RECURSIVE\_DETAILS] Script Date: 11-10-2024 16:21:01 \*\*\*\*\*\*/

SET ANSI\_NULLS ON

GO

SET QUOTED\_IDENTIFIER ON

GO

-- =============================================

-- AUTHOR: AMIT SHUDDHALWAR

-- CREATE DATE: 19-09-2024

-- DESCRIPTION: DELETE THE DETAILS OF TASK RECURSIVE DETAILS

-- =============================================

ALTER PROCEDURE [dbo].[SP\_DELETE\_TASK\_RECURSIVE\_DETAILS]

-- ADD THE PARAMETERS FOR THE STORED PROCEDURE HERE

@MKEY INT,

@LAST\_UPDATED\_BY NUMERIC

AS

BEGIN

-- SET NOCOUNT ON ADDED TO PREVENT EXTRA RESULT SETS FROM

-- INTERFERING WITH SELECT STATEMENTS.

SET NOCOUNT ON;

BEGIN TRY

IF ( (SELECT COUNT(\*)

FROM TASK\_RECURSIVE\_HDR

WHERE MKEY = @MKEY) > 0 )

BEGIN

---------------------------------------------------------------------------------------------------------------------

INSERT INTO TASK\_RECURSIVE\_HDR\_H

(TASK\_NAME,

TASK\_DESCRIPTION,

TERM,

[START\_DATE],

ENDS,

END\_DATE,

CREATED\_BY,

CREATION\_DATE,

LAST\_UPDATED\_BY,

LAST\_UPDATE\_DATE)

SELECT TASK\_NAME,

TASK\_DESCRIPTION,

TERM,

[START\_DATE],

ENDS,

END\_DATE,

CREATED\_BY,

CREATION\_DATE,

LAST\_UPDATED\_BY,

LAST\_UPDATE\_DATE

FROM TASK\_RECURSIVE\_HDR

WHERE MKEY = @MKEY;

---------------------------------------------------------------------------------------------------------------------

UPDATE TASK\_RECURSIVE\_HDR

SET DELETE\_FLAG = 'Y',

LAST\_UPDATED\_BY = @LAST\_UPDATED\_BY,

LAST\_UPDATE\_DATE = GETDATE()

WHERE MKEY = @MKEY;

---------------------------------------------------------------------------------------------------------------------

INSERT INTO TASK\_RECURSIVE\_TRL\_H

(MKEY,

TERM\_TYPE,

CREATED\_BY,

CREATION\_DATE,

LAST\_UPDATED\_BY,

LAST\_UPDATE\_DATE,

ATTRIBUTE1,

ATTRIBUTE2,

ATTRIBUTE3,

ATTRIBUTE4,

ATTRIBUTE5,

ATTRIBUTE6,

ATTRIBUTE7,

ATTRIBUTE8,

ATTRIBUTE9,

ATTRIBUTE10,

ATTRIBUTE11,

ATTRIBUTE12,

ATTRIBUTE13)

SELECT MKEY,

TERM\_TYPE,

CREATED\_BY,

CREATION\_DATE,

LAST\_UPDATED\_BY,

LAST\_UPDATE\_DATE,

ATTRIBUTE1,

ATTRIBUTE2,

ATTRIBUTE3,

ATTRIBUTE4,

ATTRIBUTE5,

ATTRIBUTE6,

ATTRIBUTE7,

ATTRIBUTE8,

ATTRIBUTE9,

ATTRIBUTE10,

ATTRIBUTE11,

ATTRIBUTE12,

ATTRIBUTE13

FROM TASK\_RECURSIVE\_TRL

WHERE MKEY = @MKEY;

---------------------------------------------------------------------------------------------------------------------

UPDATE TASK\_RECURSIVE\_TRL

SET DELETE\_FLAG = 'Y',

LAST\_UPDATED\_BY = @LAST\_UPDATED\_BY,

LAST\_UPDATE\_DATE = GETDATE()

WHERE MKEY = @MKEY;

---------------------------------------------------------------------------------------------------------------------

END

END TRY

BEGIN CATCH

SELECT ERROR\_NUMBER() AS ERRORNUMBER,

ERROR\_SEVERITY() AS ERRORSEVERITY,

ERROR\_STATE() AS ERRORSTATE,

ERROR\_PROCEDURE() AS ERRORPROCEDURE,

ERROR\_LINE() AS ERRORLINE,

ERROR\_MESSAGE() AS ERRORMESSAGE;

END CATCH

END