

0. Summary

1. Temp Table In Dynamic SQL
 2. Life scope of temp Table In Dynamic SQL
 3. Life scope of temp Table In Dynamic SQL
 4. Clean up
-

0. Summary

1.

```
--CREATE PROCEDURE spTempTableInDynamicSQL
--AS
-- BEGIN
--     DECLARE @sql NVARCHAR(MAX) = 'Create Table #TempTable(Id INT)
--         insert into #TempTable values (500)';
--     EXECUTE sp_executesql @sql;
--     SELECT *
--     FROM   #TempTable;
-- END;
--GO -- Run the previous command and begins new batch
--EXECUTE spTempTableInDynamicSQL;
```

Return Error

Temp tables created by dynamic SQL are dropped
when the dynamic SQL block in the stored procedure completes execution.
Thus, the calling procedure can not access them.

2.

```
--CREATE PROCEDURE spTempTableInDynamicSQL
--AS
-- BEGIN
--     CREATE TABLE #TempTable ( Id INT );
--     INSERT INTO #TempTable
--     VALUES ( 500 );
--     DECLARE @sql NVARCHAR(MAX) = 'SELECT * FROM #TempTable';
--     EXECUTE sp_executesql @sql;
-- END;
--GO -- Run the previous command and begins new batch
--EXECUTE spTempTableInDynamicSQL;
---- Return 500
--SELECT * FROM #TempTable;
---- Return Error
```

Dynamic SQL block can access temp tables created by the calling stored procedure
But the temp tables created by the stored procedure
are not accessible from the query out side of the stored procedure.

1. Temp Table In Dynamic SQL

```
--T044_01_Temp Table In Dynamic SQL
```

```
IF ( EXISTS ( SELECT      *
               FROM        INFORMATION_SCHEMA.ROUTINES
               WHERE        ROUTINE_TYPE = 'PROCEDURE'
                           AND LEFT(ROUTINE_NAME, 3) NOT IN ( 'sp_', 'xp_', 'ms_' )
                           AND SPECIFIC_NAME = 'spTempTableInDynamicSQL' ) )

BEGIN
    DROP PROCEDURE spTempTableInDynamicSQL;
END;

GO -- Run the previous command and begins new batch
CREATE PROCEDURE spTempTableInDynamicSQL
AS
BEGIN
    DECLARE @sql NVARCHAR(MAX) = 'Create Table #TempTable(Id INT)
                                   insert into #TempTable values (500)
                                   Select * from #TempTable';
    EXECUTE sp_executesql @sql;
END;

GO -- Run the previous command and begins new batch
EXECUTE spTempTableInDynamicSQL;

GO -- Run the previous command and begins new batch
--Return 500
```

2. Life scope of temp Table In Dynamic SQL

```
--T044_02_Life scope of temp Table In Dynamic SQL
```

```
IF ( EXISTS ( SELECT      *
               FROM        INFORMATION_SCHEMA.ROUTINES
               WHERE        ROUTINE_TYPE = 'PROCEDURE'
                           AND LEFT(ROUTINE_NAME, 3) NOT IN ( 'sp_', 'xp_', 'ms_' )
                           AND SPECIFIC_NAME = 'spTempTableInDynamicSQL' ) )

BEGIN
    DROP PROCEDURE spTempTableInDynamicSQL;
END;

GO -- Run the previous command and begins new batch
CREATE PROCEDURE spTempTableInDynamicSQL
AS
BEGIN
    DECLARE @sql NVARCHAR(MAX) = 'Create Table #TempTable(Id INT)
                                   insert into #TempTable values (500)';
    EXECUTE sp_executesql @sql;
    SELECT *
    FROM    #TempTable;
END;

GO -- Run the previous command and begins new batch
EXECUTE spTempTableInDynamicSQL;

GO -- Run the previous command and begins new batch
--Error
```

```

/*
1.
Output as following
--Msg 208, Level 16, State 0, Procedure spTempTableInDynamicSQL, Line 8 [Batch Start Line 60]
--Invalid object name '#TempTable'.
2.
#TempTable created by dynamic SQL is dropped
when the dynamic SQL block in the stored procedure completes execution.
*/

```

=====

3. Life scope of temp Table In Dynamic SQL

```

-----
--T044_03_Life scope of temp Table In Dynamic SQL
-----

```

```

IF ( EXISTS ( SELECT      *
                FROM        INFORMATION_SCHEMA.ROUTINES
                WHERE        ROUTINE_TYPE = 'PROCEDURE'
                            AND LEFT(ROUTINE_NAME, 3) NOT IN ( 'sp_', 'xp_', 'ms_' )
                            AND SPECIFIC_NAME = 'spTempTableInDynamicSQL' ) )

    BEGIN
        DROP PROCEDURE spTempTableInDynamicSQL;
    END;

GO -- Run the previous command and begins new batch
CREATE PROCEDURE spTempTableInDynamicSQL
AS
    BEGIN
        CREATE TABLE #TempTable ( Id INT );
        INSERT INTO #TempTable
        VALUES ( 500 );
        DECLARE @sql NVARCHAR(MAX) = 'SELECT * FROM #TempTable';
        EXECUTE sp_executesql @sql;
    END;

GO -- Run the previous command and begins new batch
EXECUTE spTempTableInDynamicSQL;
-- Return 500
SELECT * FROM #TempTable;

GO -- Run the previous command and begins new batch
-- Return Error
/*
1.
Output as following.
1.1.
--EXECUTE spTempTableInDynamicSQL;
Return 500
1.2.
--SELECT * FROM #TempTable;
Error message
--Msg 208, Level 16, State 0, Line 107
--Invalid object name '#TempTable'.
2.
Dynamic SQL block can access temp tables created by the calling stored procedure
But the temp tables created by the stored procedure
are not accessible from the query out side of the stored procedure.
*/

```

=====

4. Clean up

```
--=====
--T044_04_Clean up
--=====

IF ( EXISTS ( SELECT      *
               FROM        INFORMATION_SCHEMA.ROUTINES
               WHERE        ROUTINE_TYPE = 'PROCEDURE'
                           AND LEFT(ROUTINE_NAME, 3) NOT IN ( 'sp_', 'xp_', 'ms_' )
                           AND SPECIFIC_NAME = 'spTempTableInDynamicSQL' ) )

    BEGIN
        DROP PROCEDURE spTempTableInDynamicSQL;
    END;

GO -- Run the previous command and begins new batch
```