## **SQL Procedure For Finding Fill Rate in a Table - Dynamic Approach**

#### Input Parameters

Both of the Input Parameters are mandatory.

**1.** @p\_TableName - Data type used for this input Parameter is NVARCHAR(128) and Nullability is NOT NULL.

**2.** @p\_Include\_BlankAsNotFilled - Data type used for this input Parameter is BIT and Nullability is NOT NULL and either 0 or 1 needs to give. 0 is by Default and 0 means OFF. 1 is ON (when given as 1 - Blank entries will be considered As Not Filled Data).

**Output Columns**

There are Two output Columns. both of those are Non Nullable Output Columns.

1. [Column Name] - Data type used for this Output Column is sysname and Nullability is NOT NULL. All the Column Names for the user given Table Name would come as row values.

2. [Fill Rate (%)] -  Data type used for this Output Column is DECIMAL(5,2) and Nullability is NOT NULL. Values from 0.00 to 100.00 would come in result with respective Column Names.

**Info reg Stored Procedure**

* Created the store Procedure named - '***Get\_FillRate***'.
* To avoid the number of rows returned, set NOCOUNT as ON.
* Try, Catch Blocks are added for error handling's.
* To read Uncommitted Modifications, set TRANSACTION ISOLATION LEVEL as READ UNCOMMITTED.
* Parameter Sniffing Concept is also included.
* Some handling's done on the Table Name input parameters to support user typing table name formats like '.table\_name','..table\_name','...table\_name','table\_name','[table\_name]','dbo.table\_name','dbo.[table\_name]','[dbo].[table\_name]' etc.,
* Validation is included at the start, when user gives other than 'table name', stored procedure would throw 'Table not exists in this Database' as error message.
* System table named SYS.OBJECTS and SYS.COLUMNS and System View named INFORMATION\_SCHEMA.COLUMNS are used inside the stored procedure.
* ORDINAL\_POSITION from INFORMATION\_SCHEMA.COLUMNS is used, to return the result set with the same column order that the table structure already has.
* COLLATION\_NAME from INFORMATION\_SCHEMA.COLUMNS is used, to support conditions like blank is either need to consider or not, as not filled entries.
* COLUMN\_NAME from INFORMATION\_SCHEMA.COLUMNS is used, to show the final result set with respective fill rates.
* Dynamic Query is used, to support dynamic approach and this would avoid all the challenges that would come in static solutions like schema changes.
* Both Method 1(Dynamic Query with WHILE LOOP) and Method 2(Dynamic Query with UNION ALL) produces same result sets and carries same functionality where some metrics like CPU time,Elapsed Time,Logical reads that are better in Method 2.