



MAIN FEATURE TAB

Broken SQL

A scan of all views, stored procedures, functions, and triggers — identifying those that reference missing columns, dropped tables, or otherwise fail to validate.

Who

WHO IT IS FOR

Developers and DBAs who need to know which programmable objects stopped working after a schema change.

What

WHAT IT DOES

Broken SQL validates every view, procedure, function, and trigger against the live schema and surfaces the ones that no longer compile, with the failing object and the reason shown side by side.

When

WHEN TO USE IT

Run it after altering or dropping columns and tables, after a deployment, or before a release — to catch compile failures before users do.

Where

WHERE TO FIND IT

The Broken SQL tab is on the main top tab bar (a Pro feature). Start Validation runs the scan; problems are listed with their status and error text.

Why

WHY IT MATTERS

A column rename can silently break a dozen objects. Catching them in one scan avoids late-night emergency fixes and embarrassing production failures.

SCREENSHOT

McSaver Search — Broken SQL

Start Validation

Show only problems (5 problems)

100% Complete — Failed: 5 Encrypted: 0 Successful: 297 Total: 302

Drop Selected

1 selected

#	Type	Schema	Name	Validation SQL	Status / Error
13	Func	dbo	ufnLeadingZeros	ALTER Function [dbo].[ufnLeadingZeros](	Cannot ALTER 'dbo.ufnLeadingZeros' because it is being referenced by object 'Customer'.
42	Calc	HumanResources	Employee	alter table [HumanResources].[Employee] drop	*** Unable to validate this item. The index 'IX_Employee_OrganizationLevel_OrganizationNode' is dependent on column 'OrganizationLevel'. ALTER TABLE DROP COLUMN OrganizationLevel failed because one or more objects access this column.
163	Calc	Production	Document	alter table [Production].[Document] drop	*** Unable to validate this item. The index 'AK_Document_DocumentLevel_DocumentNode' is dependent on column 'DocumentLevel'. ALTER TABLE DROP COLUMN DocumentLevel failed because one or more objects access this column.
239	Calc	Sales	Customer	alter table [Sales].[Customer] drop	*** Unable to validate this item. The index 'AK_Customer_AccountNumber' is dependent on column 'AccountNumber'. ALTER TABLE DROP COLUMN AccountNumber failed because one or more objects access this column.
293	Calc	Sales	SalesOrderHeader	alter table [Sales].[SalesOrderHeader] drop	*** Unable to validate this item. The index 'AK_SalesOrderHeader_SalesOrderNumber' is dependent on column 'SalesOrderNumber'. ALTER TABLE DROP COLUMN SalesOrderNumber failed because one or more objects access this column.

```
1 ALTER Function [dbo].[ufnLeadingZeros](
2   @Value int
3 )
4 RETURNS varchar(8)
5 WITH SCHEMABINDING
6 AS
7 BEGIN
8     DECLARE @ReturnValue varchar(8);
9
10    SET @ReturnValue = CONVERT(varchar(8), @Value);
11    SET @ReturnValue = REPLICATE('0', 8 - DATALENGTH(@ReturnValue)) + @ReturnValue;
12
13    RETURN (@ReturnValue);
14 END;
```

Broken SQL as shown on mcsavertechnologies.com



Tip of the Day 34 tips

Every Tip of the Day for this tab — open them in the app from the lightbulb icon.

- VALUE

Run Broken SQL validation after schema changes to instantly find every proc, view, and function that broke — catching compile errors before they reach production and avoiding late-night emergency fixes.
- OVERVIEW

Broken SQL validation tests every stored procedure, view, function, trigger, and constraint by executing ALTER statements inside rolled-back ADO.NET transactions — no changes are persisted.
- ACTION

Click the blue "Start Validation" button to load and test every SQL object; the same button switches to "Stop Validation" while a run is in progress.
- LIMITATION

Encrypted stored procedures, views, and functions cannot be validated and are marked as "Encrypted" in the Status / Error column.
- FILTER

"Show only problems" is checked by default — uncheck it to see all validated objects including those that passed.
- NAVIGATION

Click any row to load that object's full source code into the SQL editor panel below the results grid.

SELECTION	Clicking a row also toggles its selection — the running selection count appears at the far right of the toolbar.
CLIPBOARD	Right-click any cell to copy its content to the clipboard — the Validation SQL column copies the full SQL, not the truncated preview.
COLUMN	The Validation SQL column shows the exact ALTER statement that was executed inside the rolled-back transaction to test the object.
COLUMN	The results grid has six columns — #, Type, Schema, Name, Validation SQL, and Status / Error.
COLUMN	The Status / Error column stretches to fill the remaining window width so long error messages stay readable; on a narrow window it keeps a minimum width and the grid scrolls horizontally instead.
SAFETY	Each validation runs inside an ADO.NET transaction that is always rolled back — your database objects are never modified during a run.
VISUAL	Broken objects show yellow bold text on an amber background in the Status / Error column — hover over the status to read the full error message.
PROGRESS	The progress bar, percentage, and "Validating X / Y..." text show real-time progress — click "Stop Validation" at any time to halt the run.
FEATURE	When a run finishes, the toolbar shows a summary counting Failed, Encrypted, Successful, and Total objects.
FEATURE	The "(N problems)" badge in the toolbar updates live as broken objects are found during validation.
DETECTION	Stored procedures that reference missing tables, columns, or other objects are flagged as broken with the specific SQL Server error.
DETECTION	Views that reference deleted columns or tables fail validation — the error message tells you exactly which object is missing.
DETECTION	Functions that have parameter type mismatches or reference non-existent objects are detected and reported with the inner error message.
DETECTION	Triggers attached to tables that have been structurally modified may fail validation — check the error for column or constraint references.
TECHNIQUE	Computed columns, check constraints, and defaults are validated by dropping and re-adding them inside the rolled-back transaction.
CLARIFICATION	The status "OK" means the object compiled successfully — it does not guarantee runtime correctness or that queries return expected results.
CLARIFICATION	Objects shown as "???" could not be processed for validation — typically the definition could not be converted to an ALTER statement.
CONCEPT	While a run is in progress, pending rows show "..." in the Status / Error column until each object has been checked.
ACTION	Select one or more broken rows and click "Drop Selected ..." to generate DROP scripts for review in a separate window.

WORKFLOW The Drop Selected window wraps all DROP statements in a BEGIN/COMMIT TRANSACTION — review the editable SQL, then click "Execute DROP" to actually remove the objects.

CLARIFICATION Drop Selected only acts on broken rows — if your selection contains no broken objects, a toast tells you nothing will be dropped.

ACTION Click "Show Log" at the right edge of the toolbar to open the broken-SQL change log — a running record of the DROP scripts you have executed from this tab.

FEATURE Every DROP you run from the review window is written to the change log, noting whether it succeeded and whether you edited the script before executing it.

CLARIFICATION The change log window shows the most recent entries; use its Open Full Log button to open the complete file, which reads as a near-runnable SQL script.

CLIPBOARD Right-click the "Show Log" button to copy the change log's file path to the clipboard for pasting into an editor, a shell, or SSMS.

NAVIGATION The results grid pins its column header while you scroll — the header stays put as rows scroll vertically, and header and body share one horizontal scroll so the columns stay aligned.

EXPORT The camera icon captures a screenshot of the Broken SQL tab and opens the export folder so you can grab the saved image.

NAVIGATION The tip icon opens a random tip about using the Broken SQL tab.