


# Data Management Assets

Last Modified on 07 January 2020

The Data Management assets include the items needed to gather data for actions and asserts.

Comparison Manifest

## Comparison Manifest

Icon	Description
	The <b>Comparison Manifest</b> asset is used in the grid comparison assert. This asset defines how the two grids in the assert compare to each other. First the you need to gather the left side of the comparison, then the right. Once both sides of the comparison have been gathered, you may set which columns to compare and how to compare them.

## Gather Left Columns / Gather Right Columns

SentryOne Test Element Editor

To use the comparison manifest, you will need to select the grid assets used to perform the comparison with. Start by choosing the left and right grids. Then select the 'Gather Columns' button to move to the next step.

▶ Expected Results Grid

Selected Element:

▶ Actual Results Grid

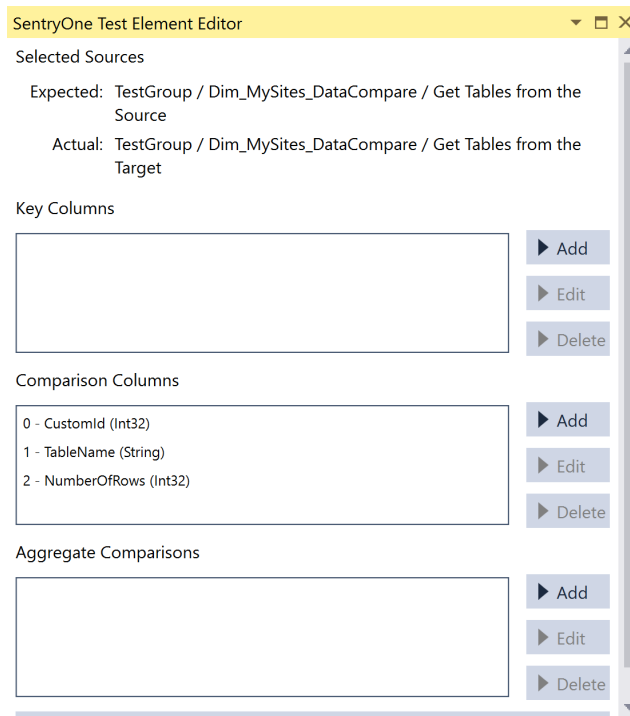
Selected Element:

▶ Gather Columns

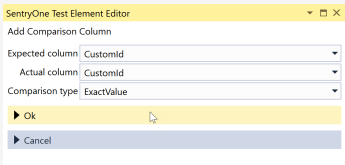
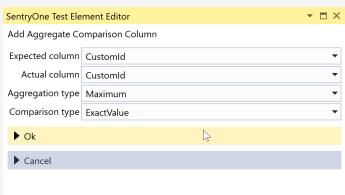
Function	Description
<b>Expected Results Grid</b>	Select the grid you wish to be the left side of the comparison.

<b>Actual Results Function</b>	Select the grid you wish to be the right side of the Comparison.
<b>Gather Columns</b>	Select Gather Columns to proceed through the Comparison Manifest.
<b>Connection</b>	The location of the data you want to compare. The connection can be set from an existing connection asset or typed into the connection string box.
<b>Query</b>	The query you want to use to gather the data you want to compare. This can be typed in the editor box, or pulled from an existing query asset.

## Comparisons




Function	Description	Image
<b>Key Columns</b>	Add, edit, or delete the key columns from each side of the comparison. Select the left and right column that match as the key column, and specify if the key is descending.	<p>The dialog box 'Add Key Column' has two dropdown menus for 'Expected column' and 'Actual column', both set to 'Customid'. Each dropdown has a checkbox for 'Is descending key'. At the bottom are 'OK' and 'Cancel' buttons.</p>

<p><b>Function</b></p> <p><b>Comparison Columns</b></p>	<p><b>Description</b></p> <p>Set the columns you want to compare. Columns with the same name and data type automatically populate to compare. In the key columns, choose a comparison type based on the type of the column. Integers have fixed tolerance options, and strings have an exact value option or a string match option with additional parameters.</p>	<p><b>Image</b></p> 
<p><b>Aggregate Comparisons</b></p>	<p>Set aggregate comparison of columns. There are several aggregation types to choose from depending on the types being compared, as well as standard choices for comparison types.</p>	

ADO.NET Connection

# ADO.NET Connection

Icon	Description
	<p>The <b>ADO.NET Connection</b> asset creates and stores a connection string to a data source. If you know the connection string, you can simply type it into the Connection string field, otherwise you can use the connection string builder.</p>

## Connection Editor

ADO.NET Connection.asset\* ✕

Provider

SqlClient Data Provider
▼

Connection string

✎ Configure

✔ Test Connection

Function	Description						
<p><b>Provider</b></p>	<p>The provider type for the connection string.</p> <p>The following providers are available:</p> <table border="1" style="width: 100%; text-align: center;"> <tbody> <tr> <td style="width: 33%;">• Odbc</td> <td style="width: 33%;">• OracleClient</td> <td style="width: 33%;">• Microsoft SQL Server Compact 4.0</td> </tr> <tr> <td>• OleDb</td> <td>• SqlClient</td> <td>• MySQL</td> </tr> </tbody> </table>	• Odbc	• OracleClient	• Microsoft SQL Server Compact 4.0	• OleDb	• SqlClient	• MySQL
• Odbc	• OracleClient	• Microsoft SQL Server Compact 4.0					
• OleDb	• SqlClient	• MySQL					

<b>Function</b> <b>Connection String</b>	<b>Description</b> The connection string for the data source.
<b>Configure</b>	Launches the connection string builder, and walks the you through creating the connection string that's stored in the asset.

## Connection Properties Window


Function	Description
<b>Data Source</b>	This is the provider type specified in the element's editor.
<b>Server Name</b>	The name of the server where the data source is located.
<b>Log on to the server</b>	Use window's authentication or provide a specific user name and password.
<b>Connect to a database</b>	Select a database located on the server specified, or attach a database file.

**Note:** Once all items are populated you can select **Test Connection** to

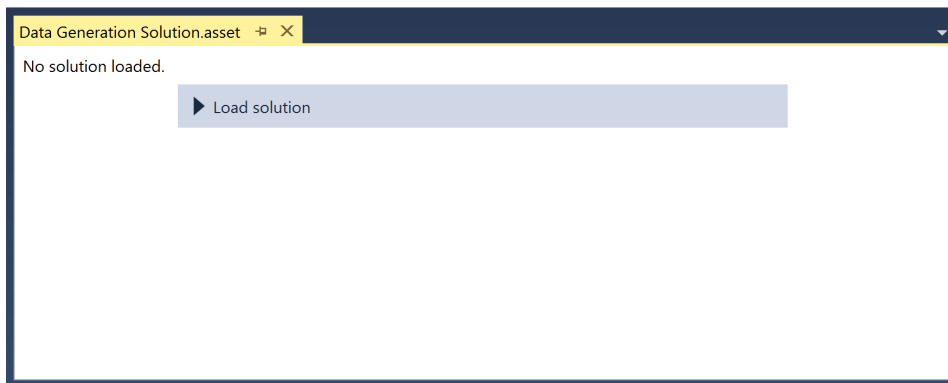
ensure that everything connects as expected. The connection populates the connection string based on the properties provided in the element's editor.

Data Generation Solution

## Data Generation Solution

Icon	Description
	The <b>Data Generation Solution</b> asset picks an existing data generation solution that was created in Pragmatic Workbench, and is later executed with the <b>Data Generation</b> action.


## Data Generation Solution Editor



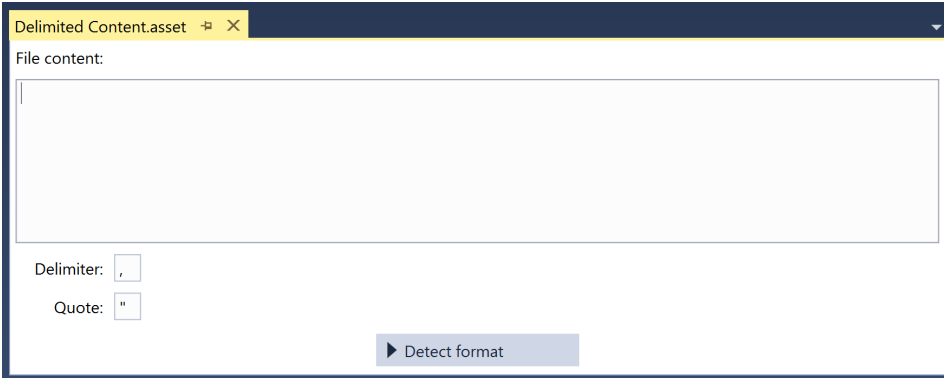
Function	Description
<b>Load Solution</b>	Select <b>Load solution</b> to open the file picker window. Browse for an existing .dgsln file and load it into the asset. Once loaded, the editor informs you about the number of tables affected.

Delimited Content

## Delimited Content

Icon	Description
	<p>The <b>Delimited Content</b> asset stores some delimited data that becomes part of the test assembly. This enables, for example, an easy set up for static data as a data driven testing source.</p>

## Initial view



Function	Description
<b>File content</b>	Type the data that you would like to store in the test assembly. This data stores in asset files and can be in any delimited format.
<b>Delimiter / Quote</b>	These are the delimiter and quote that the file has. The asset's format detection routine always checks tab, pipe and comma for delimiters and both single and double quotes for the quote. If your file uses a different delimiter or quote, you can enter them here.
<b>Detect format</b>	Begins the process of detecting the format of the file and columns.

## After scan view


Function	Description
<b>Format</b>	<p>This lists the formats that have been evaluated, along with their scores. A higher score indicates better compliance (consistent number of columns across rows and a higher number of columns in total). The format with the highest score is pre-selected; unless a custom delimiter or quote was specified, in which case the custom format is pre-selected.</p> <p><b>Note:</b> The format selection is only available immediately after running the format detection process. When coming back to edit the asset later, this option disappears.</p>
<b>Delimiter / Quote</b>	Shows the currently selected delimiter and quote.
<b>Lines to skip</b>	The number of lines of pre-amble to skip.
<b>Has header row</b>	Select this if the file has a header row containing column names. The header row comes after any pre-amble.
<b>Treat empty as NULL</b>	Select this if empty values should be rendered as values.
<b>Encoding</b>	the encoding that's used to read the file. If your file contains special characters that are represented incorrectly on preview, it's likely that the encoding needs changing.
<b>Culture</b>	This is the culture that's used to interpret the source data as typed values. Decimal points and date formats are those where the culture requires particular attention.



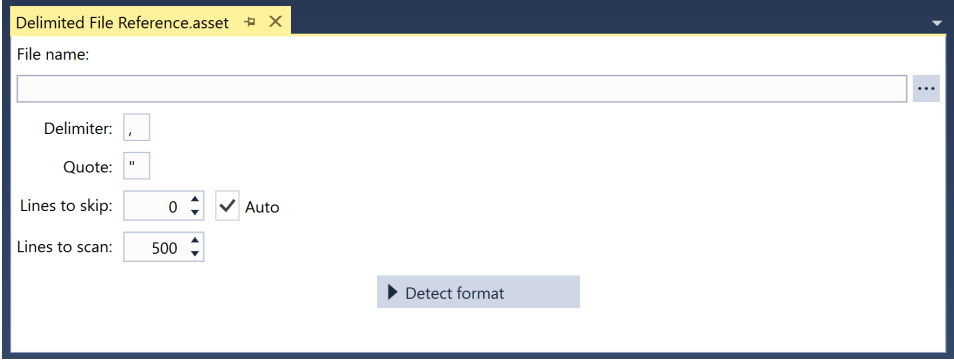
<b>Function</b> <b>Preview Data</b>	<b>Description</b> Displays the preview of the data using the selected option in a grid view.
--	--

Delimited File Reference

## Delimited File Reference

Icon	Description
	The <b>Delimited File Reference</b> asset selects a delimited file and gathers information about both its format and the data it contains.

### Initial view



Function	Description
<b>File name</b>	Paste the file name here or use the ellipsis button to select a file. A parameterized value may be used in the form %parameterName% or .
<b>Delimiter / Quote</b>	These are the delimiter and quote that the file has. The asset's format detection routine always checks tab, pipe and comma for delimiters and both single and double quotes for the quote. If your file uses a different delimiter or quote, you can enter them here.

<b>Lines to skip</b>	<b>Description</b> This is the number of rows of pre-ample to skip. Auto automatically finds out how much pre-ample there is. <b>Note:</b> Header rows are automatically detected.
<b>Lines to scan</b>	This is the number of lines of data to scan while detecting column formats.
<b>Detect format</b>	Begins the process of detecting the format of the file and columns.

## After scan view

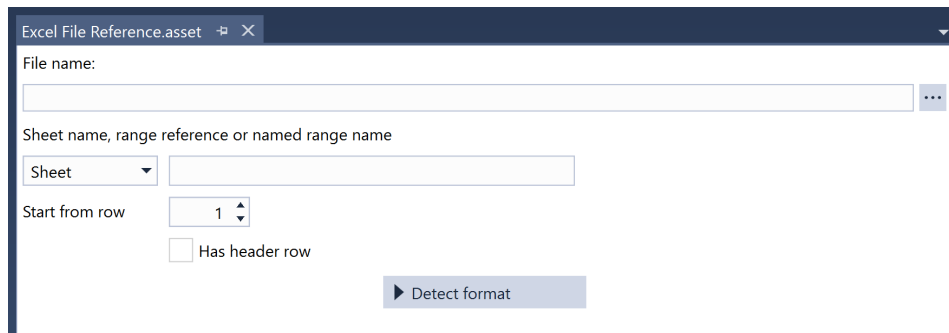
The screenshot shows a software interface for detecting file formats. The window title is "Delimited File Reference.asset\*". The "File name" field contains "C:\Users\jsindelar\Desktop\Jesse SQL Query Practice\JesseTest Query.sql". The "Format" dropdown is set to "Delimiter: Comma - Quote: Double quote - Score: 8.4". Below this, "Delimiter" is "Comma" and "Quote" is "Double quote". The "Lines to skip" is set to "0". There are checkboxes for "Has header row" and "Treat empty as NULL", both of which are unchecked. The "Encoding" is "Unicode (UTF-8)" and the "Culture" is "Invariant Language (Invariant Country)". There are buttons for "Preview Data" and "Change delimiters". The "Columns" section has a "Use same type for all" checkbox and a list of 13 columns with their detected types: Column 1: String, Column 2: String, Column 3: String, Column 4: String, Column 5: Int32, Column 6: String, Column 7: String, Column 8: Int32, Column 9: String, Column 10: String, Column 11: Int32, Column 12: String, Column 13: String.

Function	Description
<b>File name</b>	Paste the file name here or use the ellipsis button to select a file. A parameterized value may be used in the form %parameterName% or .

<b>Function</b>	<p>This lists the formats that have been evaluated, along with their scores. A higher score indicates better compliance (consistent number of columns across rows and a higher number of columns in total). The format with the highest score is pre-selected unless a custom delimiter or quote was specified, in which case the custom format is pre-selected. ⓘ</p> <p><b>Note:</b> The format selection is only available immediately after running the format detection process. When coming back to edit the asset later, this option disappears.</p>
<b>Format</b>	
<b>Delimiter / Quote</b>	Shows the currently selected delimiter and quote.
<b>Lines to skip</b>	The number of lines of pre-amble to skip.
<b>Has header row</b>	Select this if the file has a header row containing column names. The header row comes after any pre-amble.
<b>Treat empty as NULL</b>	Select this if empty values should be rendered as values.
<b>Encoding</b>	The encoding that's used to read the file. If your file contains special characters that are represented incorrectly on preview, it's likely that the encoding needs changing.
<b>Culture</b>	The culture that's used to interpret the source data as typed values. Decimal points and date formats are those where the culture requires particular attention.
<b>Preview Data</b>	Displays the preview of the data using the selected option in a grid view.
<b>Change delimiters</b>	Change the custom delimiters and run the format detection again.
<b>Column list</b>	Each column is presented along with a list of types. This list is derived by attempting format conversion on a variety of formats. You can change the resulting type to any of the allowable formats.

# Excel File Reference

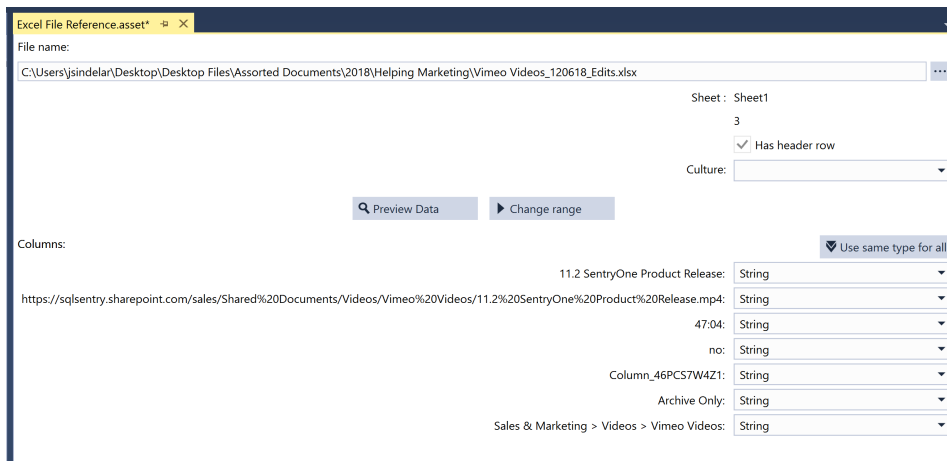
Icon	Description
	Stores an excel file as a connection asset.



The screenshot shows a configuration window titled "Excel File Reference.asset". It contains the following fields and controls:

- File name:** A text input field with a browse button (three dots).
- Sheet name, range reference or named range name:** A section containing:
  - Sheet:** A dropdown menu.
  - Sheet name, range reference or named range name:** A text input field.
- Start from row:** A numeric input field with a spinner, currently set to 1.
- Has header row:** A checkbox.
- Detect format:** A button with a right-pointing arrow.


Function	Description
<b>File Name</b>	The name of the excel file that's used for the asset connection.
<b>Sheet name, range reference, or named range name</b>	The name of the excel sheet that's used for the asset connection.
<b>Start from row</b>	Select the row where the asset starts the count.
<b>Has Header row</b>	Select whether the first row is recognized as the header row.
<b>Detect Format</b>	Begins the process of detecting the format of the columns.



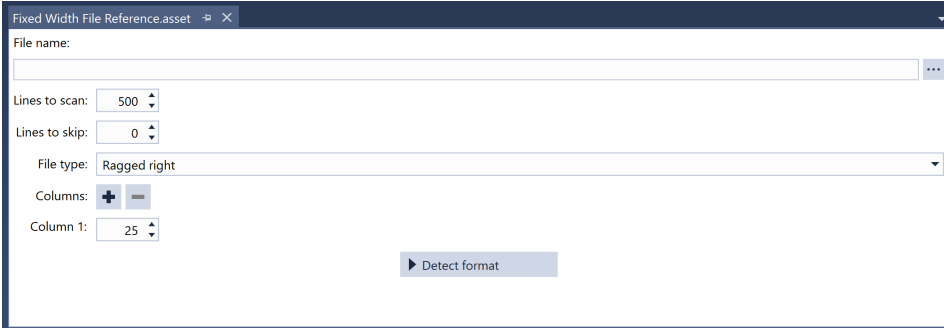
Function	Description
<b>File Name</b>	The name of the excel file that's used for the asset connection.
<b>Sheet name, range reference, or named range name</b>	The name of the excel sheet that's used for the asset connection.
<b>Start from row</b>	Select the row where the asset starts the count.
<b>Has header row</b>	Select whether the first row is recognized as the header row.
<b>Culture</b>	
<b>Preview Data</b>	Preview the selected data in the asset.
<b>Change range</b>	
<b>Columns</b>	
<b>Use same type for all</b>	

Fixed Width File Reference

## Fixed Width File Reference

Icon	Description
	<p>The <b>Fixed Width File Reference</b> asset references a fixed width file and, after specifying the column widths, determines the format data each column represents.</p>

## Initial view



Function	Description				
<p><b>File name</b></p>	<p>Paste the file name here or use the ellipsis button to select a file. A parameterized value may be used in the form %parameterName% or .</p>				
<p><b>Lines to scan</b></p>	<p>The number of lines of data to scan while detecting column formats.</p>				
	<p>The type of flat file that's being referenced. Three file types are available:</p> <table border="1" data-bbox="424 1704 1394 2067"> <thead> <tr> <th data-bbox="424 1704 746 1807">Type</th> <th data-bbox="746 1704 1394 1807">Description</th> </tr> </thead> <tbody> <tr> <td data-bbox="424 1807 746 2067"> <p><b>Ragged right</b></p> </td> <td data-bbox="746 1807 1394 2067"> <p>This is a file where the last column is not fixed width, with each line terminated by a carriage return and/or line feed. A line is parsed if it's at least as long as the sum of the widths of all but the last column.</p> </td> </tr> </tbody> </table>	Type	Description	<p><b>Ragged right</b></p>	<p>This is a file where the last column is not fixed width, with each line terminated by a carriage return and/or line feed. A line is parsed if it's at least as long as the sum of the widths of all but the last column.</p>
Type	Description				
<p><b>Ragged right</b></p>	<p>This is a file where the last column is not fixed width, with each line terminated by a carriage return and/or line feed. A line is parsed if it's at least as long as the sum of the widths of all but the last column.</p>				

<b>File type Function</b>	<b>Description</b>	<b>Description</b>
	<b>Fixed width with line breaks</b>	This is a file where lines terminated by a carriage return and/or line feed. A line is parsed if it's at least as long as the sum of the widths of all columns.
	<b>Fixed width without line breaks</b>	This is a file where there are no line breaks, and the data comes in a continuous stream. Every x characters are parsed where x is the sum of the widths of all columns.
<b>Columns</b>	<p>This is the list of column widths that is parsed from the file. You can use the + and - buttons to add or remove columns to/from the end of the list. Alternatively, you can place the caret in the first column width box, type the width and then press enter. This adds a new column where you can repeat the same process. <b>i</b></p> <p><b>Note:</b> Pressing enter in a width box only adds a new column if you are currently in the last box. Pressing enter on any other boxes moves focus to the next box.</p>	
<b>Detect format</b>	Begins the process of detecting the format of the columns.	


## After scan view

Function	Description
<b>File name</b>	Paste the file name here or use the ellipsis button to select a file. A parameterized value may be used in the form %parameterName% or .

<b>Function Has header row</b>	<b>Description</b> Select this if the file has a header row containing column names. The header row comes after any pre-amble.
<b>Treat empty as NULL</b>	Select this if empty values should be rendered as values.
<b>Encoding</b>	The encoding that's used to read the file. If your file contains special characters that are represented incorrectly on preview, it's likely that the encoding needs changing.
<b>Culture</b>	The culture that's used to interpret the source data as typed values. Decimal points and date formats are those where the culture requires particular attention.
<b>Preview Data</b>	Displays the preview of the data using the selected option in a grid view.
<b>Change delimiters</b>	Change the custom delimiters and run the format detection again.
<b>Column list</b>	Each column is presented along with a list of types. This list is derived by attempting format conversion on a variety of formats. You can change the resulting type to any of the allowable formats.

Grid Collection

## Grid Collection

Icon	Description
	The <b>Grid Collection</b> asset stores the details of a grid that can be used over and over using the <b>Load Grid from Asset</b> action. Supply a connection string and a query that pulls the information for the grid. Both the connection and the query can be typed or pulled from another asset.



# Grid Collection Editor




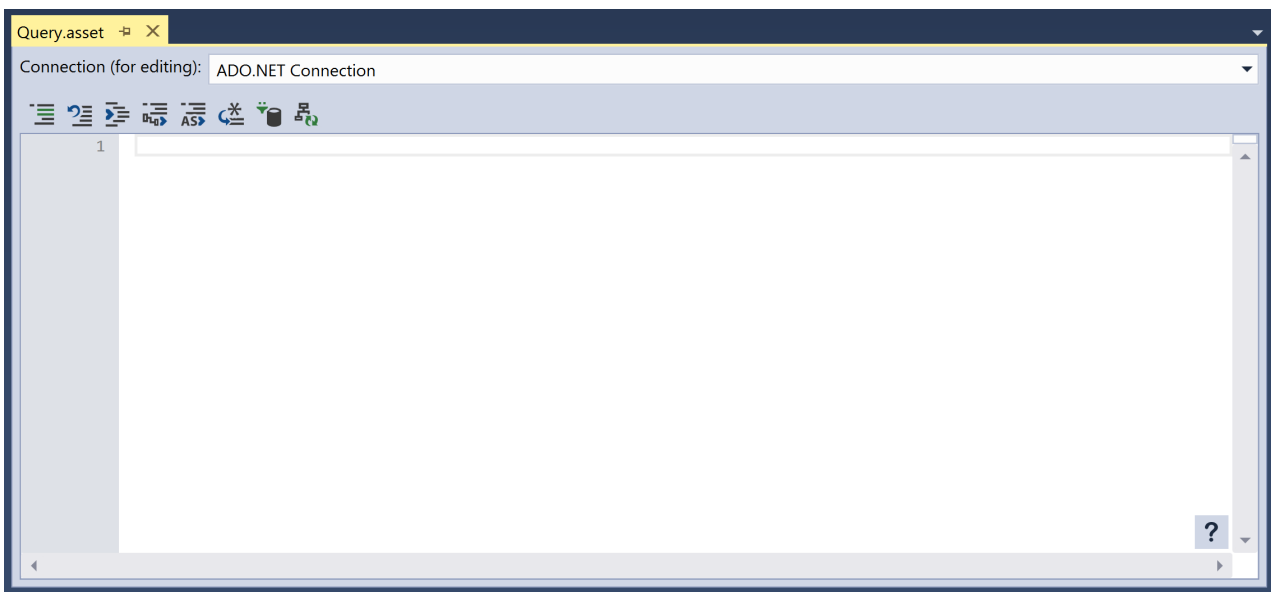
Function	Description
<b>Connection</b>	Connection string to the data source, can either be typed manually or pulled from an existing connection asset.
<b>Copy From Asset</b>	Copy the connection string data from the connection asset.
<b>Connection String</b>	Connection string to the data source, can either be typed manually or pulled from an existing connection asset.
<b>Low memory mode</b>	Toggles whether the grid is stored on disk or in memory.
<b>Is Disk Based</b>	Store the query results within a temporary file instead of in memory. This option is best used for larger data sets.
<b>Query</b>	The query responsible for gathering the data for the grid.

Query

## Query

Icon	Description

<p><b>Icon</b></p> 	<p><b>Description</b></p> <p>The <b>Query</b> asset stores queries.</p>
--	---



When connecting to a SQL Server or Azure SQL instance, the query editor includes an advanced editing experience. You can select the connection from the drop-down at the top of the editor, which loads information about the schema of the database where the query runs. The selection is remembered for the duration of the session.

When connecting to a SQL Server or Azure SQL instance, the query editor provides the following features:


Feature	Description
<b>Code completion</b>	Suggestion lists after . or when Ctrl+Space is pressed.
<b>Quick info</b>	Hover over an object name for information about that object.
<b>Refactoring</b>	Tracked renaming of variables and temporary objects, query formatting, object qualification.
	Shows the parameters expected for procedure and function

Parameter Information	calls. Description
<b>Column choosers</b>	Select table that you want to resolve an ambiguous column.
<b>Synonym resolution</b>	Work with synonyms with the full IntelliSense experience.
<b>Shortcuts</b>	e.g. 'ssf' -> 'SELECT * FROM', st100 -> 'SELECT TOP 100 * FROM'.
<b>Tab expansion</b>	Select <b>tab</b> when the caret is to the right of a * to specify all columns.

**Note:** A summary of the available functions can be found by selecting the ? icon in the bottom right hand corner of the editor.

Validation Manifest

## Validation Manifest

Icon	Description
	The <b>Validation Manifest</b> selects a grid, and then add rules to validate the content of the grid. There are different available rules for any given type available in the grid.

Validation Manifest Editor

SentryOne Test Element Editor

Grid Asset  
 TestGroup / Dim\_MySites\_DataCompare / Get Tables from the Source

Only pass validation if no rows are present

Validation Columns

2	NumberOfRows	Greater Than	1000
1	TableName	Exact Value	

+ New  
 Edit  
 Remove

Function	Description
<b>Select Grid</b>	<p>When the editor first opens, the only available option is the <b>Select Grid</b> button which displays all available grids in the test to validate. Once a grid is selected, then the ability to add validation columns will be available.</p> <p><b>Note:</b> If you want to validate the fact that the grid is empty (for example, because your query only returns rows that fail) you can select the <b>Only pass validation if no rows are present</b> option.</p>
<b>Remove</b>	Deletes the selected rule from the manifest.
<b>New</b>	Expands the editor to show the column selector and comparison type picker.

SentryOne Test Element Editor

Grid Asset  
 TestGroup / Dim\_MySites\_DataCompare / Get Tables from the Source

Only pass validation if no rows are present

Validation Columns

2	NumberOfRows	Greater Than	1000
1	TableName	Exact Value	

+ New  
 Edit  
 Remove

Column: 0 - CustomId (Int32)

Comparison Type: Exact Value

Is Aggregated

Comparison Value:

+ Add X Cancel

<b>Function</b>	<b>Description</b>
<b>Column</b>	Selects a column from the grid to apply the comparison.
<b>Comparison Type</b>	The available comparison types vary depending on the type of the selected column. The comparison type uses the Comparison Value field to execute the validation.
<b>Is Aggregated</b>	If selected, and the column's type allows for aggregation, then another field allows you to select the type of aggregation.
<b>Comparison Value</b>	The comparison value is compared to the row of each column using the previously selected Comparison Type.
<b>Cancel</b>	Cancel out of the new comparison without saving.
<b>Add</b>	Enables once all fields are populated for the new rule and saves the rule to the Validation Manifest.