<https://learn.microsoft.com/en-us/sql/ssdt/debugger/run-transact-sql-debugger?view=sql-server-ver16>

**Run the Transact-SQL debugger**

* Article
* 08/29/2024
* 1 contributor

Feedback

**In this article**

1. [Starting and stopping the debugger](https://learn.microsoft.com/en-us/sql/ssdt/debugger/run-transact-sql-debugger?view=sql-server-ver16#starting-and-stopping-the-debugger)
2. [Controlling the debugger](https://learn.microsoft.com/en-us/sql/ssdt/debugger/run-transact-sql-debugger?view=sql-server-ver16#controlling-the-debugger)
3. [Related content](https://learn.microsoft.com/en-us/sql/ssdt/debugger/run-transact-sql-debugger?view=sql-server-ver16#related-content)

**Applies to:**  [SQL Server](https://learn.microsoft.com/en-us/sql/sql-server/sql-docs-navigation-guide?view=sql-server-ver16#applies-to)

You can start the Transact-SQL debugger after you open a Database Engine Query Editor window. You can set options to customize how the debugger runs and run your Transact-SQL code in debug mode until you stop the debugger.

**Starting and stopping the debugger**

The requirements to start the Transact-SQL debugger are as follows:

* If the Database Engine Query Editor is connected to an instance of the Database Engine on another computer, you must configure the debugger for remote debugging. For more information, see [Configure firewall rules before running the Transact-SQL debugger](https://learn.microsoft.com/en-us/sql/ssdt/debugger/configure-firewall-rules-before-running-tsql-debugger?view=sql-server-ver16).
* The Database Engine Query Editor window must be connected by using either a Windows Authentication or SQL Server Authentication login that is a member of the sysadmin fixed server role.
* The Database Engine Query Editor window must be connected to an instance of the SQL Server Database Engine. You can't run the debugger when the Query Editor window is connected to an instance that is in single-user mode.

We recommend debugging Transact-SQL code on a test server, not a production server, for the following reasons:

* Debugging is a highly privileged operation. Therefore, only members of the sysadmin fixed server role are allowed to debug in SQL Server.
* Debugging sessions often run for long periods of time while you investigate the operations of several Transact-SQL statements. Locks, such as update locks, that are acquired by the session might be held for extended periods, until the session is ended or the transaction committed or rolled back.

Starting the Transact-SQL debugger puts the Query Editor window into debug mode. When the Query Editor window enters debug mode, the debugger pauses at the first line of code. You can then step through the code, pause the execution on specific Transact-SQL statements, and use the debugger windows to view the current execution state. You can start the debugger by selecting the **Debug** button on the **Query** toolbar or selecting **Start Debugging** on the **Debug** menu.

The Query Editor window stays in debug mode until either the last statement in the Query Editor window finishes or you stop debug mode. You can stop debug mode and statement execution by using any one of the following methods:

* On the **Debug** menu, select **Stop Debugging**.
* On the **Debug** toolbar, select the **Stop Debugging** button.
* On the **Query** menu, select **Cancel Executing Query**.
* On the **Query** toolbar, select the **Cancel Executing Query** button.

You can also stop debug mode and allow for the remaining Transact-SQL statements to finish executing by selecting **Detach All** on the **Debug** menu.

**Controlling the debugger**

You can control how the Transact-SQL debugger operates by using the following menu commands, toolbars, and shortcuts:

* The **Debug** menu and the **Debug** toolbar. Both the **Debug** menu and **Debug** toolbar are inactive until the focus is placed in an open Query Editor window. They remain active until the current project is closed.
* The debugger keyboard shortcuts.
* The Query Editor shortcut menu. The shortcut menu is displayed when you right-click a line in a Query Editor window. When the Query Editor window is in debug mode, the shortcut menu displays debugger commands that apply to the selected line or string.
* Menu items and context commands in the windows that are opened by the debugger, such as the **Watch** or **Breakpoints** windows.

The following table shows the debugger menu commands, toolbar buttons, and keyboard shortcuts.

Expand table

| **Debug menu command** | **Editor shortcut command** | **Toolbar button** | **Keyboard shortcut** | **Action** |
| --- | --- | --- | --- | --- |
| **Windows/Breakpoints** | Not available | **Breakpoints** | CTRL+ALT+B | Display the **Breakpoints** window in which you can view and manage breakpoints. |
| **Windows/Watch/Watch1** | Not available | **Breakpoints/Watch/Watch1** | CTRL+ALT+W, 1 | Display the **Watch1** window. |
| **Windows/Watch/Watch2** | Not available | **Breakpoints/Watch/Watch2** | CTRL+ALT+W, 2 | Display the **Watch2** window. |
| **Windows/Watch/Watch3** | Not available | **Breakpoints/Watch/Watch3** | CTRL+ALT+W, 3 | Display the **Watch3** window. |
| **Windows/Watch/Watch4** | Not available | **Breakpoints/Watch/Watch4** | CTRL+ALT+W, 4 | Display the **Watch4** window. |
| **Windows/Locals** | Not available | **Breakpoints/Locals** | CTRL+ALT+V, L | Display the **Locals** window. |
| **Windows/Call Stack** | Not available | **Breakpoints/Call Stack** | CTRL+ALT+C | Display the **Call Stack** window. |
| **Windows/Threads** | Not available | **Breakpoints/Threads** | CTRL+ALT+H | Display the **Threads** window. |
| **Continue** | Not available | **Continue** | ALT+F5 | Run to the next breakpoint. **Continue** isn't active until you're focused on a Query Editor window that is in debug mode. |
| **Start Debugging** | Not available | **Start Debugging** | ALT+F5 | Put a Query Editor window into debug mode and run to the first breakpoint. If you're focused on a Query Editor window that is in debug mode, **Start Debugging** is replaced by **Continue**. |
| **Break All** | Not available | **Break All** | CTRL+ALT+BREAK | This feature not used by the Transact-SQL debugger. |
| **Stop Debugging** | Not available | **Stop Debugging** | SHIFT+F5 | Take a Query Editor window out of debug mode and return it to regular mode. |
| **Detach All** | Not available | Not available | Not available | Stops debug mode, but executes the remaining statements in the Query Editor window. |
| **Step Into** | Not available | **Step Into** | F11 | Run the next statement, and also open a new Query Editor window in debug mode if the next statement runs a stored procedure, trigger, or function. |
| **Step Over** | Not available | **Step Over** | F10 | Same as **Step Into**, except that no functions, stored procedures, or triggers are debugged. |
| **Step Out** | Not available | **Step Out** | SHIFT+F11 | Execute the remaining code in a trigger, function, or stored procedure without pausing for any breakpoints. Regular debug mode resumes when control is returned to the code that called the module. |
| Not available | **Run To** Cursor | Not available | CTRL+F10 | Execute all code from the last stop location to the current cursor location without stopping at any breakpoints. |
| **QuickWatch** | **QuickWatch** | Not available | CTRL+ALT+Q | Display the **QuickWatch** window. |
| **Toggle Breakpoint** | **Breakpoint/Insert Breakpoint** | Not available | F9 | Position a breakpoint on the current or selected Transact-SQL statement. |
| Not available | **Breakpoint/Delete Breakpoint** | Not available | Not available | Delete the breakpoint from the selected line. |
| Not available | **Breakpoint/Disable Breakpoint** | Not available | Not available | Disable the breakpoint on the selected line. The breakpoint remains on the line of code, but execution does not stop until it's reenabled. |
| Not available | **Breakpoint/Enable Breakpoint** | Not available | Not available | Enable the breakpoint on the selected line. |
| **Delete All Breakpoints** | Not available | Not available | CTRL+SHIFT+F9 | Delete all breakpoints. |
| **Disable All Breakpoints** | Not available | Not available | Not available | Disable all breakpoints. |
| Not available | **Add Watch** | Not available | Not available | Add the selected expression to the **Watch** window. |

**Related content**

* [Transact-SQL Debugger](https://learn.microsoft.com/en-us/sql/ssdt/debugger/transact-sql-debugger?view=sql-server-ver16)
* [Run the Transact-SQL Debugger](https://learn.microsoft.com/en-us/sql/ssdt/debugger/run-transact-sql-debugger?view=sql-server-ver16)
* [Step Through Transact-SQL Code](https://learn.microsoft.com/en-us/sql/ssdt/debugger/step-through-transact-sql-code?view=sql-server-ver16)
* [Debug Stored Procedures](https://learn.microsoft.com/en-us/sql/ssdt/debugger/debug-stored-procedures?view=sql-server-ver16)
* [Transact-SQL Debugger Information](https://learn.microsoft.com/en-us/sql/ssdt/debugger/transact-sql-debugger-information?view=sql-server-ver16)
* [Live Query Statistics](https://learn.microsoft.com/en-us/sql/relational-databases/performance/live-query-statistics?view=sql-server-ver16)