

Integrated terminal in SQL Operations Studio (preview)

In SQL Operations Studio (preview), you can open an integrated terminal, starting from the root of the current workspace.

In SQL Operations Studio (preview), you can open an integrated terminal, starting from the root of the current workspace. This can be convenient because you do not have to switch the windows or change the status of the current terminal to execute the command line task quickly.

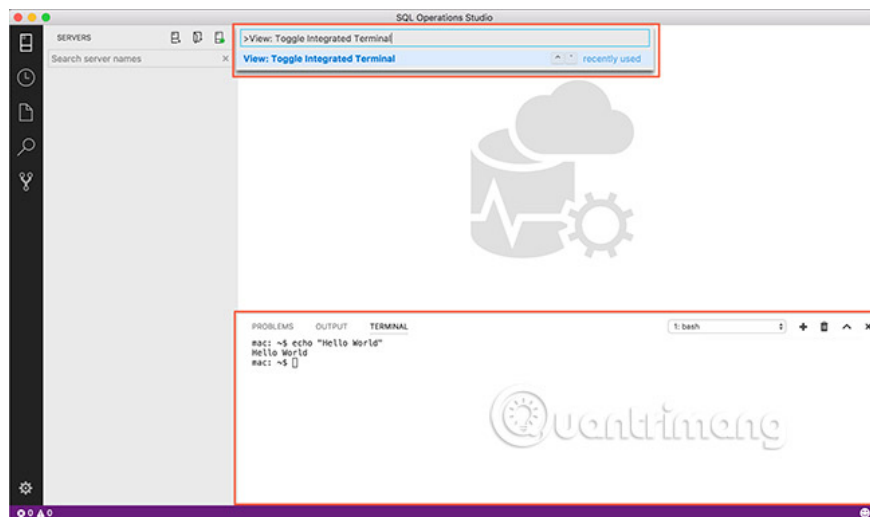
Integrated terminal in SQL Operations Studio (preview)

1. Open the terminal
2. Manage multiple terminals
3. Configuration
 1. Windows
 2. Argument Shell
4. Settings show terminal
5. Bind the keys in the terminal
 1. Run the selected text
 2. Copy and paste
 3. Search
 4. Change the name of the terminal session
 5. Forced to assign the right keys over the terminals

Open the terminal

To open the terminal:

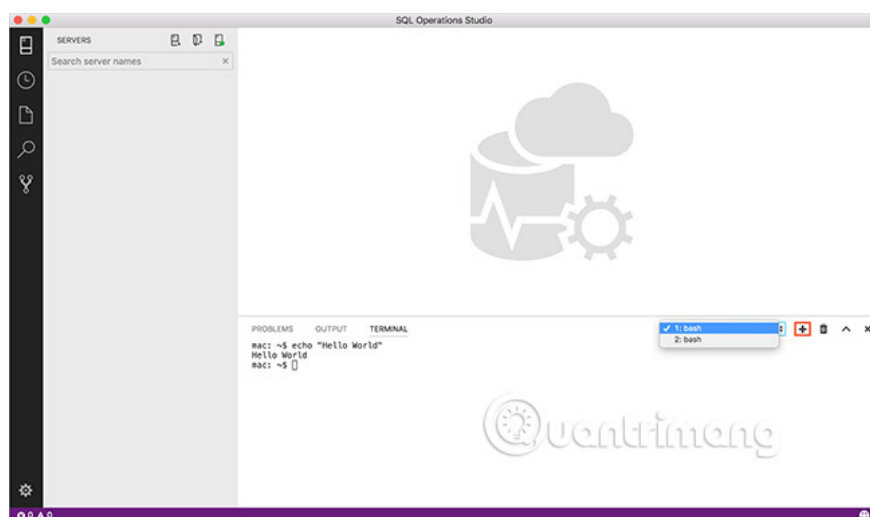
1. Use keyboard shortcut **Ctrl + `** .
2. Use the **View** | command **Integrated Terminal** menu .
3. From the **Command Palette (Ctrl + Shift + P)** , use the **View** command : **Toggle Integrated Terminal**.



Note : You can still open an external shell with Explorer Open in Command Prompt (open in Terminal on Mac or Linux), if you prefer to work outside of SQL Operations Studio (preview).

Manage multiple terminals

You can create multiple terminals in different locations and easily navigate between them. You can add terminals by clicking the plus icon at the top right of the **TERMINAL** control panel or by activating the **Ctrl + Shift + `** command. This creates another entry in the list, which can be used to switch between terminals.



Delete the terminal by pressing the **Trash** button .

Tip: If you use multiple terminals, you can add the main constraints for **focusNext**, **focusPrevious** and **kill** commands listed in the " **Binding keys in the terminal** " section below, to allow navigation between the terminals with only keyboard.

Configuration

Shell is used by default as \$ SHELL on Linux and macOS, PowerShell on Windows 10 and cmd.exe on previous versions of Windows. They can be manually overwritten, by setting the **terminal.integrated.shell**. * In the settings. The arguments can be transferred to terminal shell on Linux and macOS, using the **terminal.settings.integrated.shellArgs**. * .

Windows

Properly configure your shell on Windows including locating the appropriate executable file and updating the settings. Below is a list of common shell executables and their default locations:

```
// 64-bit cmd if available, otherwise 32-bit
"terminal.integrated.shell.windows": "C: Windowssysnativecmd.exe"
// 64-bit PowerShell if available, otherwise 32-bit
"terminal.integrated.shell.windows": "C: WindowssysnativeWindowsPowerShellv1.0
// Git Bash
"terminal.integrated.shell.windows": "C: Program FilesGitbinbash.exe"
// Bash on Ubuntu (on Windows)
"terminal.integrated.shell.windows": "C: Windowssysnativebash.exe"
```

Note : To be used as an integrated terminal, the executable shell must be a console application so **stdin / stdout / stderr** can be redirected.

Tip : Shell integrated terminal is running with the rights of SQL Operations Studio (preview). If you need to run shell commands with advanced admin rights or different permissions, you can use platform utilities like **runas.exe** in a terminal.

Argument Shell

You can pass arguments to the shell when it is started.

For example, to allow bash to run as a login shell (run **.bash_profile**), switch in the **-l** argument (with quotation marks):

```
// Linux
"terminal.integrated.shellArgs.linux": ["-l"]
```

Settings show terminal

You can customize the font and line height for the integrated terminal with the following settings:

1. `terminal.integrated.fontFamily`
2. `terminal.integrated.fontSize`
3. `terminal.integrated.lineHeight`

Bind the keys in the terminal

View command : Toggle Integrated Terminal is linked with **Ctrl + `** to quickly switch the terminal control panel built-in and out of sight.

Here are the shortcuts for quick navigation in the integrated terminal:

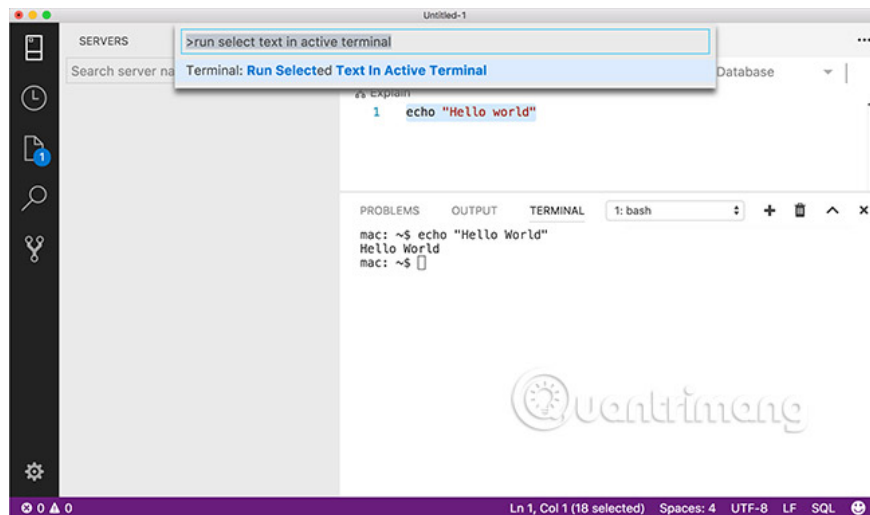
Command key combination Ctrl + ` Integrated terminal display Ctrl + Shift + ` Create new terminal Ctrl + Up Scroll up Ctrl + Down Scroll down Ctrl + PageUp Scroll page Ctrl + PageDown Scroll down page Ctrl + Home Scroll to the top of the page Ctrl + End Scroll to the bottom of the page Ctrl + K Delete the terminal

Other terminal commands are available and can be assigned with your favorite shortcuts, including:

1. **workbench.action.terminal.focus:** Focus on the terminal. This is like a conversion but focuses on the terminal instead of hiding them, if the user can see them.
2. **workbench.action.terminal.focusNext:** Focus on the next terminal instance.
3. **workbench.action.terminal.focusPrevious:** Focus the previous instance instance.
4. **workbench.action.terminal.kill:** Remove the current terminal instance.
5. **workbench.action.terminal.runSelectedText:** Run the selected text in the terminal.
6. **workbench.action.terminal.runActiveFile:** Run the active file in the terminal.

Run the selected text

To use the **runSelectedText** command , select the text in an editor and run the **Terminal** command : **Run Selected Text in Active Terminal** through the **Command Palette (Ctrl + Shift + P)** . Terminal will run the selected text:



If no text is selected in the active editor, the line that the cursor is set to is run in the terminal.

Copy and paste

Assigning keys to copy and paste conform to platform standards:

1. Linux: **Ctrl + Shift + C** and **Ctrl + Shift + V**
2. Mac: **Cmd + C** and **Cmd + V**
3. Windows: **Ctrl + C** and **Ctrl + V**

Search

The integrated terminal has a basic search function that can be activated with **Ctrl + F**.

If you want **Ctrl + F** to go to the shell, instead of launching the search utility on Linux and Windows, you need to delete the key link as follows:

```
{ "key": "ctrl + f", "command": "-workbench.action.terminal.focusFindWidget",  
  "when": "terminalFocus" },
```

Change the name of the terminal session

Integrated terminal sessions can be renamed using the **Terminal** command : **Rename** (**workbench.action.terminal.rename**) . The new name is displayed in the terminal selection menu.

Forced to assign the right keys over the terminals

While the focus is on the integrated terminal, many of the key keys assigned will not work, because key combinations are transmitted and are suppressed by the terminal itself. `Terminal.integrated.commandsToSkipShell` setting can be used to do this. It contains an array of command names with assigned keys, bypassing shell processing and instead is handled by the system binding SQL Operations Studio (preview) keys. By default, this includes all the keys assigned in the terminal, adding some commonly used key options.

See more:

1. Check constraints in SQL Server
2. Unique constraints in SQL Server
3. How to completely uninstall SQL Server

You finished reading the article "**Integrated terminal in SQL Operations Studio (preview)**" edited by the [TipsMake](#) team. We hope this article has provided you with many useful tech tips and tricks. You can search for similar articles on tips and guides. Thank you for reading and for following us regularly.