Using Command Line Options

You can start Qt Creator and specify some options from the command line. For example, you can open a file to any line and column.

To specify command line options, enter the following command in the Qt Creator installation or build directory:

```
qtcreator [option] [filename[:line_number[:column_number]]]
```

> **Note:** You can use either a colon (:) or a plus sign (+) as a separator between the filename and line number and the line number and the column number. You can also use a space between the separator and the line number.

For example, on Windows:

- `C:\qtcreator\bin>qtcreator -help`
- `C:\qtcreator\bin>qtcreator C:\TextFinder\textfinder.cpp:100:2`
- `C:\qtcreator\bin>qtcreator C:\TextFinder\textfinder.cpp +100+2`

On macOS:

- `Qt\ Creator.app/Contents/MacOS/Qt\ Creator -help`

To open a project that is located in a particular folder, you can pass on the folder name as a command line argument. Qt Creator looks for a session that matches the folder name and loads it. Or it looks for a project file in the folder and opens it. For example:

```
qtcreator .
```

> **Note:** To run a self-built Qt Creator from the command line on Windows, make sure that the Qt installation directory is included in the PATH environment variable. You can enter the following command on the command line to add Qt to the path:

```
set PATH=<Qt_installation_directory>\mingw\bin;c:<Qt_installation_directory>\bin;%PATH%
```

The following table summarizes the available options:

Option	Description

-client	Attempt to connect to an already running instance of Qt Creator.
-pid	Attempt to connect to an already running instance of Qt Creator with the specified process ID.
-block	Open files in editors in a running Qt Creator instance and block the command line until the first editor is closed.
-nocrashcheck	Disable the startup check for a previously crashed Qt Creator instance.
-load <plugin>	Enable the specified plugin and all plugins that it depends on. You can combine `-load` and `-noload` options and specify both options multiple times to enable and disable several plugins. The actions are executed in the specified order.
-load all	Enable all plugins.
-noload <plugin>	Disable the specified plugin and all plugins that depend on it.
-noload all	Disable all plugins.
-profile	Output profiling data about plugin startup and shutdown.
-pluginpath <path>	Add a path where Qt Creator looks for plugins. To specify several paths, add the `-pluginpath` option for each path.
-settingspath <path>	Override the default path where user settings are stored.
-installsettingspath <path>	Override the default path from where user-independent settings are read (for example written by the installer).
-temporarycleansettings, -tcs	Use clean settings for debug or testing reasons. The settings will be deleted when Qt Creator exits.
-test <plugin>[,testfunction[:testdata]]...	For Qt Creator plugin developers: run the plugin's tests using a separate settings path by default.
-test all	For Qt Creator plugin developers: run tests from all plugins.
-notest <plugin>	For Qt Creator plugin developers: exclude all of the plugin's tests from the test run.
-scenario <scenarioname>	For Qt Creator plugin developers: run the specified scenario.
-color <color>	Core plugin: override the selected UI color.
-presentationMode	Core plugin: display keyboard shortcuts as popups when you press them. Mostly useful when presenting Qt Creator to someone else.
-theme <default | dark>	Core plugin: apply a dark color theme to Qt Creator, without using stylesheets.
-notour	Welcome plugin: skip the UI tour on startup.
-debug <pid>	Debugger plugin: attach to the process with the given process ID.
-debug <executable>[,kit=<kit>]	Debugger plugin: launch and debug the executable with the name `executable`. A `kit` can be specified by ID or name to point to non-default debuggers and sysroots.
-debug [executable,]core=<corefile>[,kit=<kit>]	Debugger plugin: load the core file named `corefile`. The parameter `executable` specifies the executable that produced the core file. If this parameter is omitted, Qt Creator will attempt to reconstruct it from the core file itself. This will fail for paths with more than about 80 characters. In such cases the `executable` parameter is mandatory. A `kit` can be specified by ID or name to point to non-default debuggers
Option	**Description**

<executable>,server=<server:port>[,kit=<kit>]	`server`. The parameter `executable` specifies a local copy of the executable the remote debug server is manipulating. A `kit` can be specified by ID or name to point to non-default debuggers and sysroots.
-wincrashevent <event-handle:pid>	Debugger plugin: attach to crashed processes by using the specified event handle and process ID.
-git-show <git commit hash>	Git plugin: show the specified commit hash.
-customwizard-verbose	ProjectExplorer plugin: display additional information when loading custom wizards. For more information about custom wizards, see Adding New Custom Wizards
-ensure-kit-for-binary <path to binary>	ProjectExplorer plugin: create a kit with a toolchain corresponding to the given binary's architecture.
-lastsession	ProjectExplorer plugin: load the last session when Qt Creator starts. Open the projects and files that were open when you last exited Qt Creator. For more information about managing sessions, see Managing Sessions.
<session>	ProjectExplorer plugin: load the given session when Qt Creator starts. Open the projects and files that were open when you last exited Qt Creator. For more information about managing sessions, see Managing Sessions.

Using Custom Styles

Qt Creator is a Qt application, and therefore, it accepts the command line options that all Qt applications accept. For example, you can use the `-style` and `-stylesheet` options to apply custom styles and stylesheets. The styling is only applied during the current session.

Exercise caution when applying styles, as overriding the existing styling may make some items difficult to see. Also, setting a stylesheet may affect the text editor color scheme and the styling of the integrated Qt Designer.

You can also switch to a dark theme to customize the appearance of widgets, colors, and icons without using stylesheets.

‹ Setting Up Conan

Keyboard Shortcuts ›