

## 使用命令行选项

您可以启动 Qt 创建器并从命令行指定一些选项。例如，可以将文件打开到任何行和列。

要指定命令行选项，请在 Qt Creator 安装或构建目录中输入以下命令：

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

**注意：** 您可以使用冒号 (:)或加号 (+) 作为文件名和行号以及行号和列号之间的分隔符。还可以在分隔符和行号之间使用空格。

例如，在窗口上：

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

在苹果操作系统上：

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

若要打开位于特定文件夹中的项目，可以将文件夹名称作为命令行参数传递。Qt创建器查找与文件夹名称匹配的会话并加载它。或者，它在文件夹中查找项目文件并将其打开。例如：

```
qtcreeator .
```

**注意：** 要从 Windows 上的命令行运行自建的 Qt 创建器，请确保 Qt 安装目录包含在 PATH 环境变量中。您可以在命令行上输入以下命令以将Qt添加到路径中：

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

下表总结了可用的选项：

选择	描述: _____
-帮助	在命令行选项上显示帮助。
-版本 选择	显示Qt创建者版本。 描述: _____

-块	在正在运行的 Qt Creator 实例中的编辑器中打开文件并阻止命令行，直到第一个编辑器关闭。
-无崩溃检查	禁用以前崩溃的 Qt 创建者实例的启动检查。
-负载<>	启用指定的插件及其依赖的所有插件。您可以组合 和 选项并多次指定这两个选项，以启用和禁用多个插件。操作按指定的顺序执行。-load-noload
-全部加载	启用所有插件。
-空载<>	禁用指定的插件以及依赖它的所有插件。
-全部不加载	禁用所有插件。
-轮廓	输出有关插件启动和关闭的分析数据。
-插件路径<路径>	添加Qt创建者查找插件的路径。若要指定多个路径，请为每个路径添加选项。-pluginpath
-设置路径<路径>	覆盖存储用户设置的默认路径。
-安装设置路径<路径>	覆盖从中读取与用户无关的设置的默认路径（例如，由安装程序写入）。
-临时清洁剂，-tcs	出于调试或测试原因使用干净的设置。当Qt创建者退出时，设置将被删除。
-测试<>[, 测试功能[: 测试数据]]...	对于 Qt Creator 插件开发者：默认情况下，使用单独的设置路径运行插件的测试。
-全部测试	对于Qt创建者插件开发人员：从所有插件运行测试。
- <>	对于Qt Creator插件开发人员：从测试运行中排除插件的所有测试。
-场景<场景>	对于 Qt 创建器插件开发者：运行指定的场景。
-颜色<颜色>	核心插件：覆盖选定的UI颜色。
-演示模式	核心插件：按下键盘快捷键时，将其显示为弹出窗口。在向其他人展示Qt创建者时最有用。
-主题<默认 黑暗>	核心插件：将深色主题应用于Qt创建器，而无需使用样式表。
-诺图尔	欢迎插件：在启动时跳过UI导览。
-调试<皮德>	调试器插件：附加到具有给定进程ID的进程。
-调试<可执行>[, 套件=<套件>]	调试器插件：启动并调试名为 的可执行文件。可以通过 ID 或名称指定，以指向非默认调试器和 sysroot。executablekit
-debug [executable,]core=<corefile>[,kit=<kit>]	Debugger plugin: load the core file named . The parameter 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 parameter is mandatory. A can be specified by ID or name to point to non-default debuggers and sysroots.corefileexecutableexecutablekit
-debug <executable>,server=<server:port>[,kit=<kit>]	Debugger plugin: attach to a debug server running on the port on the server . The parameter specifies a local copy of the executable the remote debug server is manipulating. A can be specified by ID or name to point to non-default debuggers and sysroots.portserverexecutablekit
-wincrashevent <event-handle:pid>	Debugger plugin: attach to crashed processes by using the specified event handle and process ID.
-git-show <git	Git plugin: show the specified commit hash. -----

选择

描述:

verbose	more information about custom wizards, see <a href="#">Adding New Custom Wizards</a>
-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 <a href="#">Managing Sessions</a> .
<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 <a href="#">Managing Sessions</a> .

## 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 and options to apply custom styles and [stylesheets](#). The styling is only applied during the current session. - style - stylesheet

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 >](#)

© 2022 The Qt Company Ltd. Documentation contributions included herein are the copyrights of their respective owners. The documentation provided herein is licensed under the terms of the [GNU Free Documentation License version 1.3](#) as published by the Free Software Foundation. Qt and respective logos are trademarks of The Qt Company Ltd in Finland and/or other countries worldwide. All other trademarks are property of their respective owners.



Contact Us

### Company

- About Us
- Investors
- Newsroom
- Careers
- Office Locations

### Licensing

- Terms & Conditions
- Open Source
- FAQ



Support Services  
Professional Services  
Partners  
Training

Support Center  
Downloads  
Qt Login  
Contact Us  
Customer Success

Community

Contribute to Qt  
Forum  
Wiki  
Downloads  
Marketplace

© 2022 The Qt Company

[Feedback](#) [Sign In](#)