Qt **DOCUMENTATION**

Search   Qt Creator Manual 8.0.2

Topics >

Qt Creator Manual   >   Configuring Qt Creator

Configuring Qt Creator

If you install Qt Creator as part of a Qt installation, you should be able to use it out-of-the-box, with the default settings and configuration options.

However, if you install the stand-alone Qt Creator package, build Qt Creator from sources, or install several Qt versions, you may need to tell Qt Creator where to find the Qt versions and compilers by adding the paths to them and by creating kits that use them.

To make Qt Creator behave more like your favorite code editor or IDE, you can change the settings for keyboard shortcuts, color schemes, generic highlighting, code snippets, and version control systems. In addition, you can enable and disable Qt Creator features by managing plugins.

The following sections summarize the options that you have and point you to detailed information to help you specify any required settings and to make using Qt Creator a better experience for you.

Checking Build and Run Settings

Qt Creator is an integrated development environment (IDE) that you can use to develop Qt applications. While you can use the Qt Installer to install Qt Creator, the stand-alone Qt Creator installer never installs Qt or any Qt tools, such as qmake. To use Qt Creator for Qt development, you also need to have a Qt version and a compiler installed. If you update the compiler version later, you can register it into Qt Creator.

The Qt Installer attempts to auto-detect the installed compilers and Qt versions. If it succeeds, the relevant kits will automatically become available in Qt Creator. If it does not, you must add the kits yourself to tell Qt Creator where everything is.

To add kits, select **Edit** > **Preferences** > **Kits** > **Add**.

For more information, see Adding Kits.

Each kit consists of a set of values that define one environment, such as a device, compiler, and Qt version. If you know you have installed a Qt version, but it is not listed in **Edit** > **Preferences** > **Kits** > **Qt Versions**, select **Link with Qt**.

If the Qt version is still not listed under **Auto-detected**, select **Add** to add it manually.

For more information, see Adding Qt Versions.

Also check that your compiler is listed in **Edit** > **Preferences** > **Kits** > **Compilers**.

For more information, see Adding Compilers.

You can connect devices to the development PC to run, debug, and analyze applications on them from Qt Creator. You can connect the device to the development PC via USB. Additionally, you can connect Linux-based devices over a WLAN. You must also configure a connection between Qt Creator and the development PC and specify the device

DOCUMENTATION

For more information, see Connecting Devices.

Changing Keyboard Shortcuts

You can use Qt Creator with your favorite keyboard shortcuts.

To view and edit all keyboard shortcuts defined in Qt Creator, select **Edit** > **Preferences** > **Environment** > **Keyboard**. For more information, see Keyboard Shortcuts.

Changing Color Schemes

Themes enable you to customize the appearance of the Qt Creator UI: widgets, colors, and icons.

To switch themes, select **Edit** > **Preferences** > **Environment**, and then select a theme in the **Theme** field.

You can use the Qt Creator text and code editors with your favorite color scheme that defines how code elements are highlighted and which background color is used. You can select one of the predefined color schemes or create custom ones. The color schemes apply to highlighting C++ files, QML files, and generic files.

To change the color scheme, select **Edit** > **Preferences** > **Text Editor** > **Fonts & Color**.

For more information, see Defining Color Schemes.

Generic highlighting is provided by KSyntaxHighlighting, which is the syntax highlighting engine for Kate syntax definitions. Qt Creator comes with most of the commonly used syntax files, and you can download additional files.

To download and use highlight definition files, select **Edit** > **Preferences** > **Text Editor** > **Generic Highlighter**.

For more information, see Generic Highlighting.

Adding Your Own Code Snippets

As you write code, Qt Creator suggests properties, IDs, and code snippets to complete the code. It provides a list of context-sensitive suggestions to the statement currently under your cursor. You can add, modify, and remove snippets in the snippet editor.

To open the snippet editor, select **Edit** > **Preferences** > **Text Editor** > **Snippets**.

For more information, see Editing Code Snippets.

Configuring Version Control Systems

Qt Creator supports several version control systems. In most cases, you do not need to configure the version control in any special way to make it work with Qt Creator. Once it is set up correctly on the development PC, it should just work.

However, some configuration options are available and you can set them in **Edit** > **Preferences** > **Version Control** > **General**.

For more information about the supported functions, see Using Version Control Systems.

Qt DOCUMENTATION

Qt Creator comes with a set of plugins, some of which are disabled by default. You can enable disabled plugins if you need them and disable plugins you don't need.
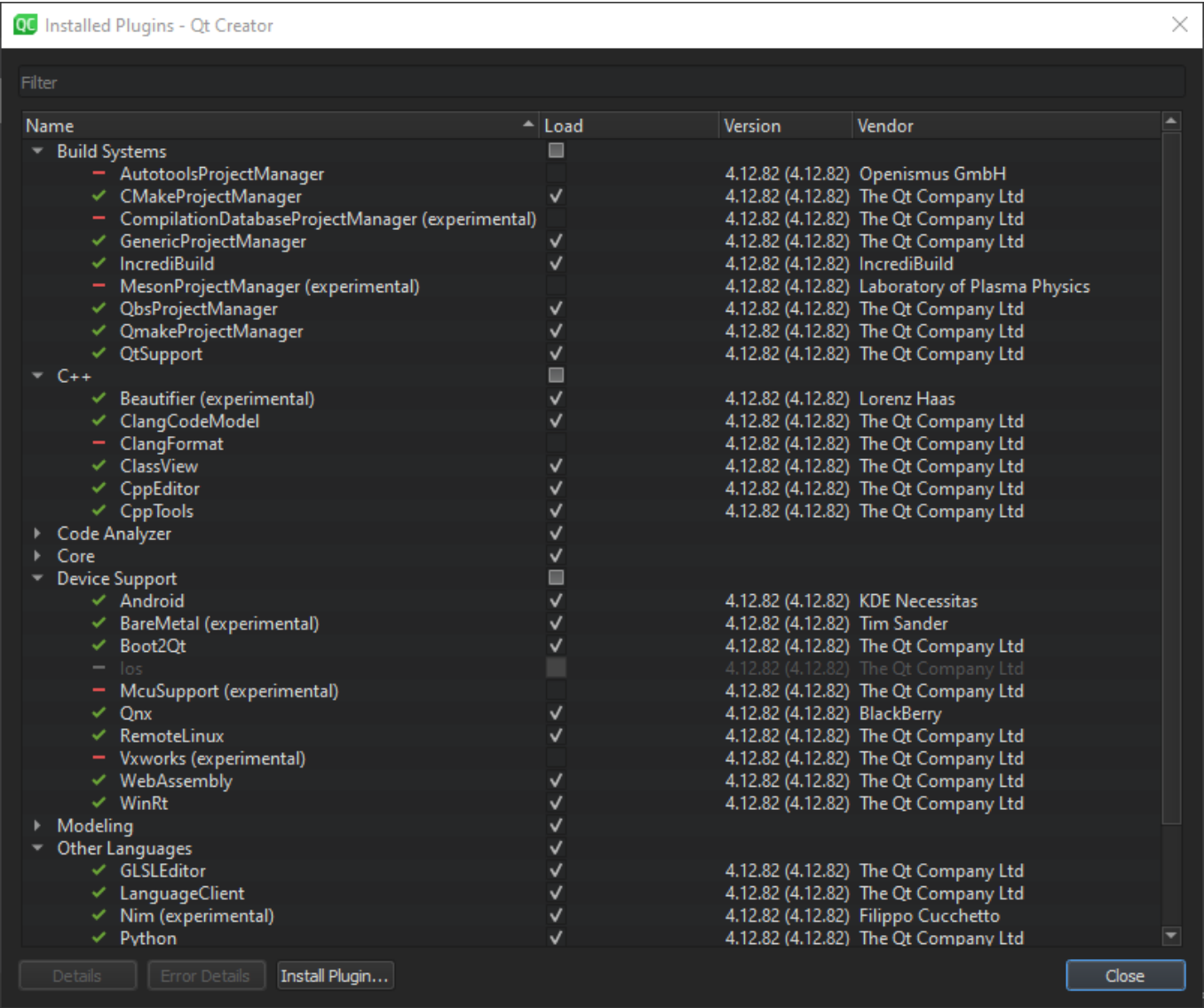
You can also install additional plugins that you downloaded from Qt Marketplace or some other source, such as GitHub.

Enabling and Disabling Plugins

New Qt Creator plugins are often introduced as *experimental plugins* to let you try them out before they are fully supported. Experimental plugins are disabled by default and you must enable them for them to become visible after you restart Qt Creator. By default, all the plugins that the plugin depends on are also enabled.

You can also disable plugins that you do not use, to streamline Qt Creator. By default, all the plugins that depend on the plugin are also disabled.

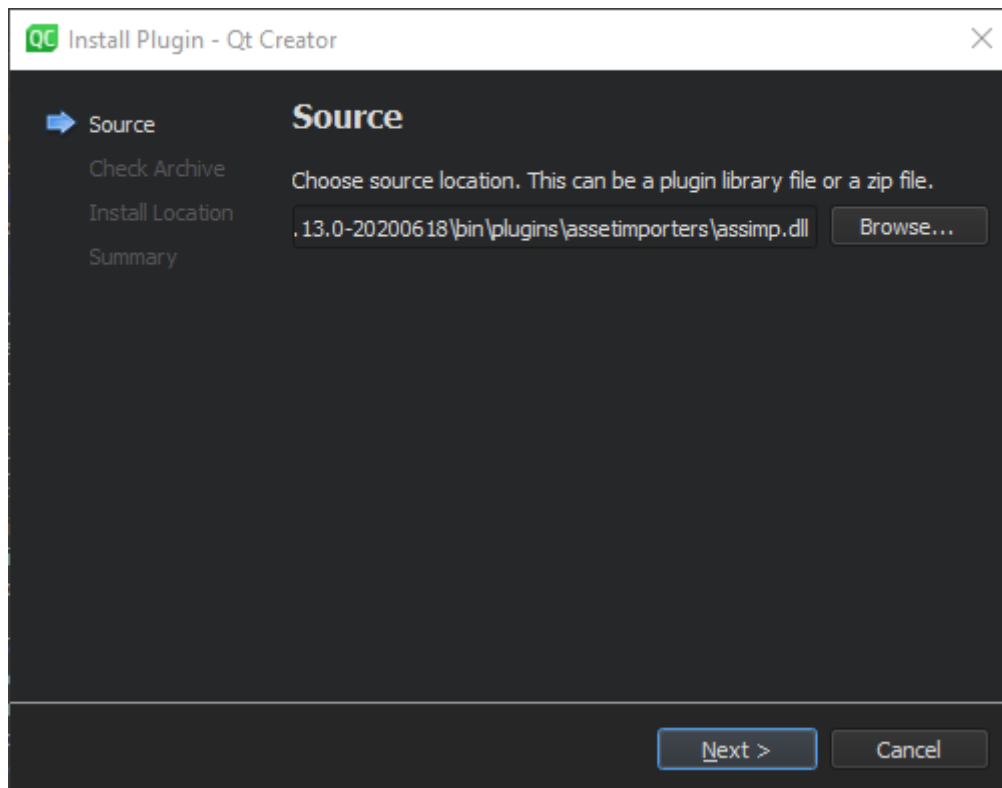To enable and disable plugins, select **Help** > **About Plugins**.
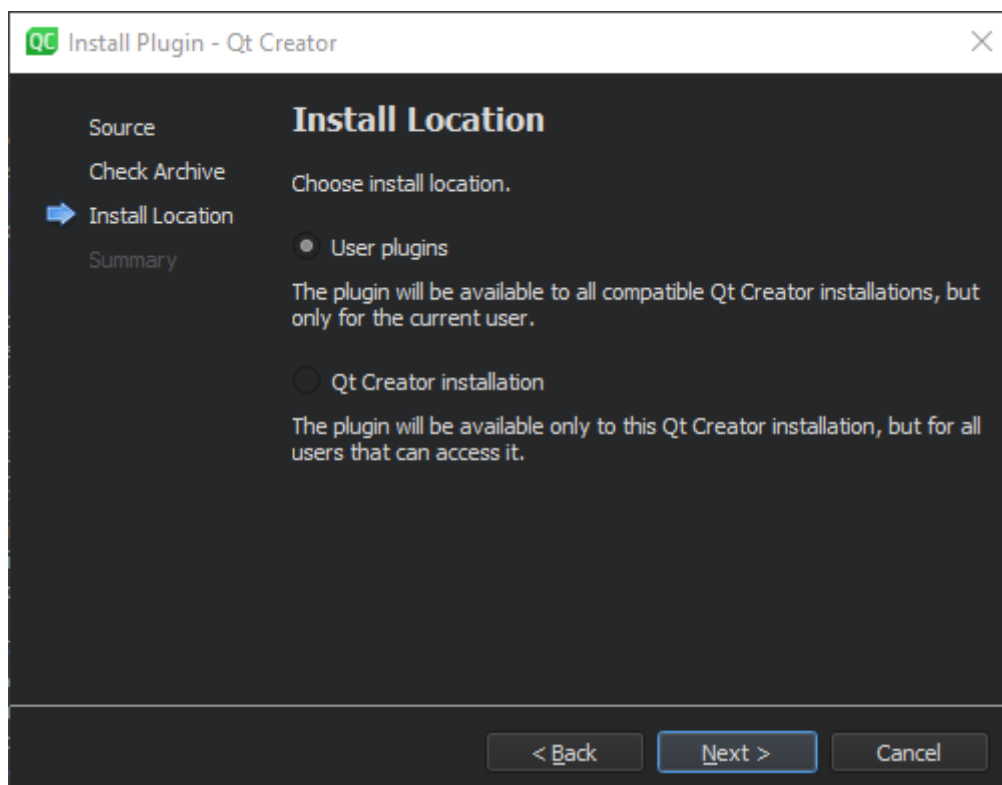

Installing Plugins

Qt Marketplace contains links to Qt Creator plugins that you can download and install either for free or for a price set by their publisher. You can browse the available plugins in the **Marketplace** tab in the Welcome mode.
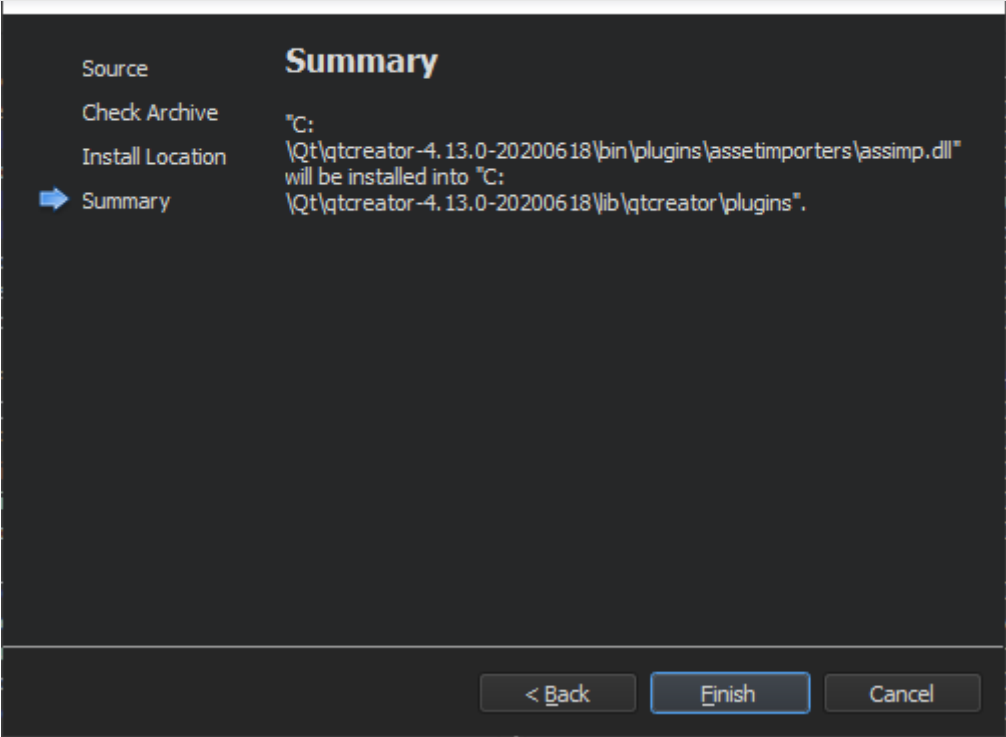
Qt DOCUMENTATION

To install plugins:

1. Select **Help** > **About Plugins** > **Install Plugins**.

2. In the **Source** dialog, enter the path to the archive or library that contains the plugin.

3. In the **Install Location** dialog, select **User plugins** to make the plugin available for the current user in all compatible Qt Creator instances or **Qt Creator installation** to make the plugin available for all users of a particular Qt Creator instance.

4. In the **Summary** dialog, select **Finish** to install the plugin.

5. Select **Restart Now** to restart Qt Creator and load the plugin.

<Viewing Output   Building and Running an Example >

© 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   **For Customers**

Partners

Training

Qt Login

Contact Us

Customer Success

Community

Contribute to Qt

Forum

Wiki

Downloads

Marketplace

Feedback   Sign In