

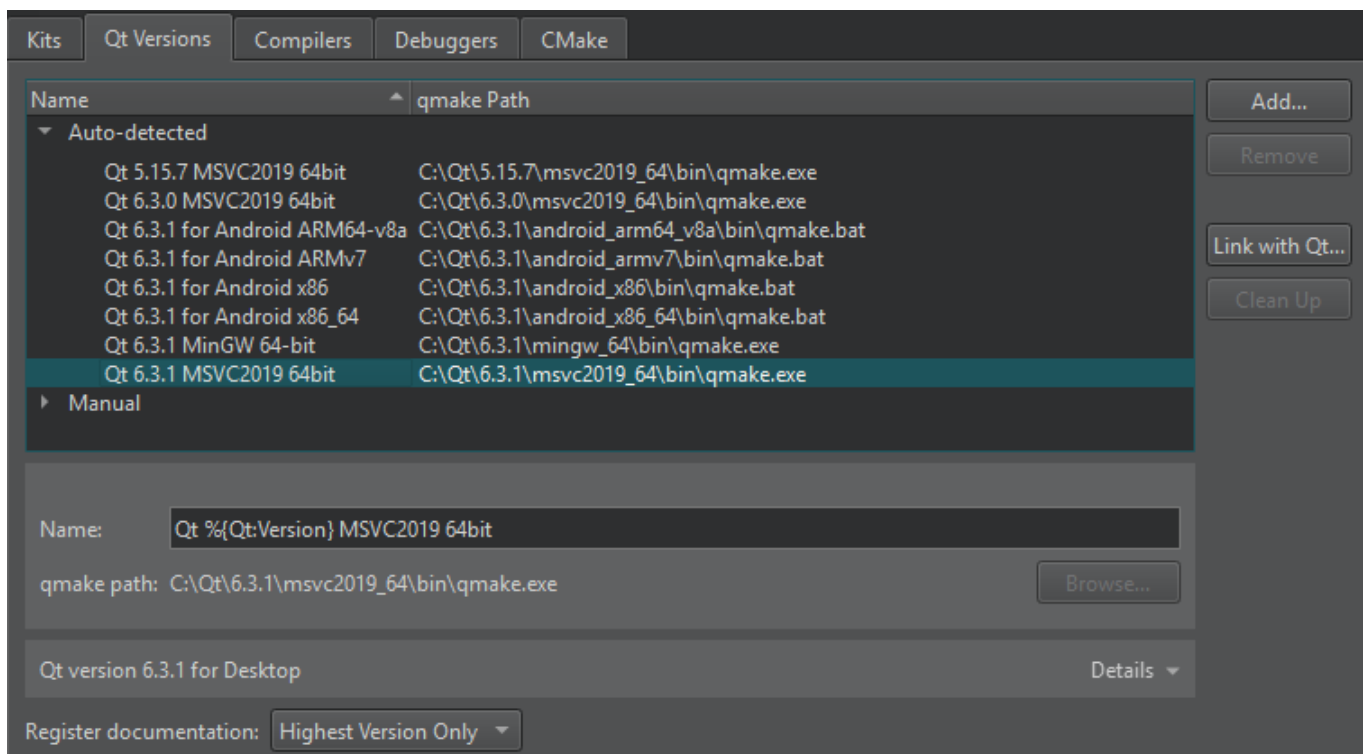
[Qt 创建者手册](#) > [添加 Qt 版本](#)

添加 Qt 版本

Qt Creator 允许您在开发 PC 上安装多个版本的 Qt，并使用不同的版本来构建您的项目。例如，[设备制造商](#)提供特殊的 Qt 版本，用于为其设备开发应用程序。

注册已安装的 Qt 版本

Qt 创建器会自动检测由您的系统或 Qt 安装程序注册的 Qt 版本。要查看每个 Qt 版本的详细信息，请在列表中选择它，然后在 **Qt 版本** 部分中选择 **详细信息**。

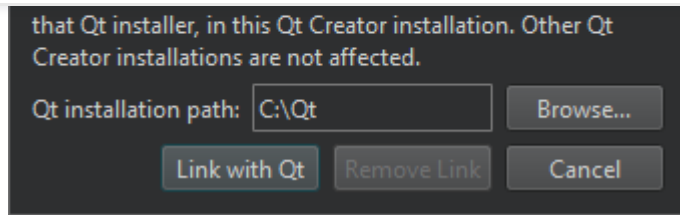


若要删除无效的 Qt 版本，请选择“**清理**”。

您可以链接到使用 Qt 维护工具安装的 Qt，以自动注册已安装的 Qt 版本。但是，您不能链接到系统使用其他软件包管理器（例如您的 Linux 发行版，在 macOS 上酿造或 Windows 上的巧克力）安装的 Qt 版本，也不能链接到自建 Qt。在这些情况下，按照设置新 **Qt 版本** 中的说明，在“Qt 版本”选项卡中选择“**添加**”以手动添加 **Qt 版本**。

要链接到 Qt 安装：

1. 选择“**编辑**>**首选项**>”（或在 macOS 上选择 **Qt 创建者**>**首选项**）>**套件**>**Qt 版本**>与 Qt 链接”。



2. 在 **Qt 安装路径** 字段中，输入安装 Qt 的目录的路径。
3. 选择“**与 Qt 链接**”以在 Qt 安装目录中自动注册 Qt 版本和工具包。
4. 选择立即**重新启动**以重新启动 Qt 创建者。

若要从列表中删除自动检测到的 Qt 版本，请选择“**删除链接**”。

如果 Qt 版本仍未列在“**自动检测**”下的“**Qt 版本**”选项卡中，则必须手动进行设置，如下一节所述。

您可以在**编辑>首选项>工具包**中指定用于构建和运行项目的每个**工具包**的 Qt 版本，如**指定工具包设置**中所述。

设置新的 Qt 版本

要添加 Qt 版本：

1. 选择“**编辑>首选项>套件>Qt 版本**”>添加”。
2. 选择要添加的 Qt 版本的 qmake 可执行文件。
3. 选择 Qt 版本以查看和编辑它。
4. 在“**名称**”字段中，编辑 Qt 创建者为 Qt 版本建议的名称。
5. 在“**qmake 路径**”字段中，可以更改 qmake 位置。
6. 如果 Qt 版本适用于 QNX，请在 QNX 开发工具包字段中输入已安装的 **QNX 开发工具包** 的路径。

若要删除手动添加的 Qt 版本，请在“**手动**”列表中选择该版本，然后选择“**删除**”。

Registering Documentation

By default, Qt Creator registers only the latest available version of the documentation for each installed Qt module.

To register the documentation sets of all installed Qt versions, choose **All** from the **Register documentation** list. To register no Qt documentation at all, choose **None**. The default behavior is **Highest Version Only**.

Troubleshooting Qt Installations

If Qt Creator detects problems in the installation of a Qt version, it displays warnings and errors beside the name of the Qt version in the list. Select the Qt version to see more information about the issue and suggestions for fixing it.

To determine whether a particular Qt version is installed correctly, Qt Creator calls and checks that the directories referenced in the output exist. When Qt Creator complains about the installation of a self-built Qt version, try running in the build directory to actually install Qt into the configured location. If you installed Qt using the Qt Installer, run the Qt maintenance tool to check for updates or to reinstall the Qt version.
`qmake -query make install`



If your build of Qt is incomplete but you still want to use Qmake as build system, you need to ensure the following minimum requirements to be able to use that setup with Qt Creator.

1. qmake is an executable that understands the command line argument `-query`
2. The `bin` and `include` directories have to exist. Qt Creator fetches these directories by running `bin/includeqmake -query`
3. The directory should be complete enough to parse `.pro` files.`mkspecs`

If your Qt version has no `ABI.LibQtCore.so`, Qt Creator is unable to detect the ABI.

[< Adding Kits](#)[Adding Compilers >](#)

© 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

[Support Services](#)
[Professional Services](#)
[Partners](#)
[Training](#)

For Customers

[Support Center](#)
[Downloads](#)
[Qt Login](#)
[Contact Us](#)
[Customer Success](#)

Community

[Contribute to Qt](#)



Downloads

Marketplace

© 2022 The Qt Company

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