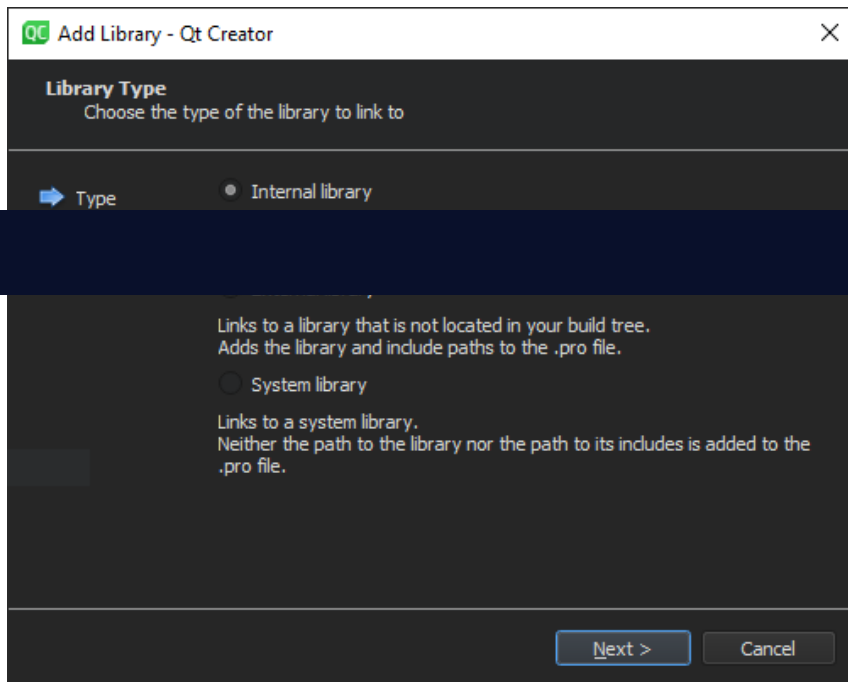


[Qt创作者手册](#) > [向项目添加库](#)

向项目添加库

除了Qt库之外，您还可以将其他库添加到项目中。库的添加方式取决于库的类型和位置。您可以添加系统库、您自己的库或第三方库。该库可以位于当前项目的构建树中，也可以位于另一个构建树中。

[Qt创作者手册8.0.2](#)
[Topics](#) >

由于系统库通常不会更改，并且通常在缺省情况下找到，因此在添加库时无需指定库或其包含的路径。可用于在编译期间查询系统库。`pkg-config`

对于您自己的库和第三方库，您需要指定路径。Qt Creator尝试猜测外部库的包含路径，但您需要检查它并在必要时对其进行修改。Qt Creator会自动为内部库添加包含路径。

对于所有库，请选择应用程序、库或插件的目标平台。

指定库是静态链接还是动态链接。对于静态链接的内部库，Qt Creator在项目文件中添加依赖项（使用CMake时`target_link_libraries`，使用qmake时`PRE_TARGETDEPS`）。

根据开发平台的不同，可能会自动检测某些选项。例如，在 macOS 上，系统会自动检测资源库类型（“资源库”或“框架”），并且该选项处于隐藏状态。但是，如果您在 macOS 以外的其他平台上进行开发，并且想要为 macOS 构建项目，则必须指定资源库类型。

Windows 上的默认约定是库的调试版本和发布版本具有相同的名称，但放在在不同的子目录中，通常称为调试和发布。如果库路径不包含这些文件夹中的任何一个，则无法选择将库放置在单独文件夹中的选项。

或者，可以将字母 *d* 添加到调试版本的库名称中。例如，如果发布版本称为 `example.lib`，则调试版本称为 `exampled.lib`。您可以指定为调试版本添加该字母，为发布版本删除该字母。如果库名称以 *d* 结尾，请取消选择“删除发行版本的“d”后缀”选项。

Qt Creator支持在项目成功构建并链接到它们后对添加的库进行代码完成和语法突出显示。

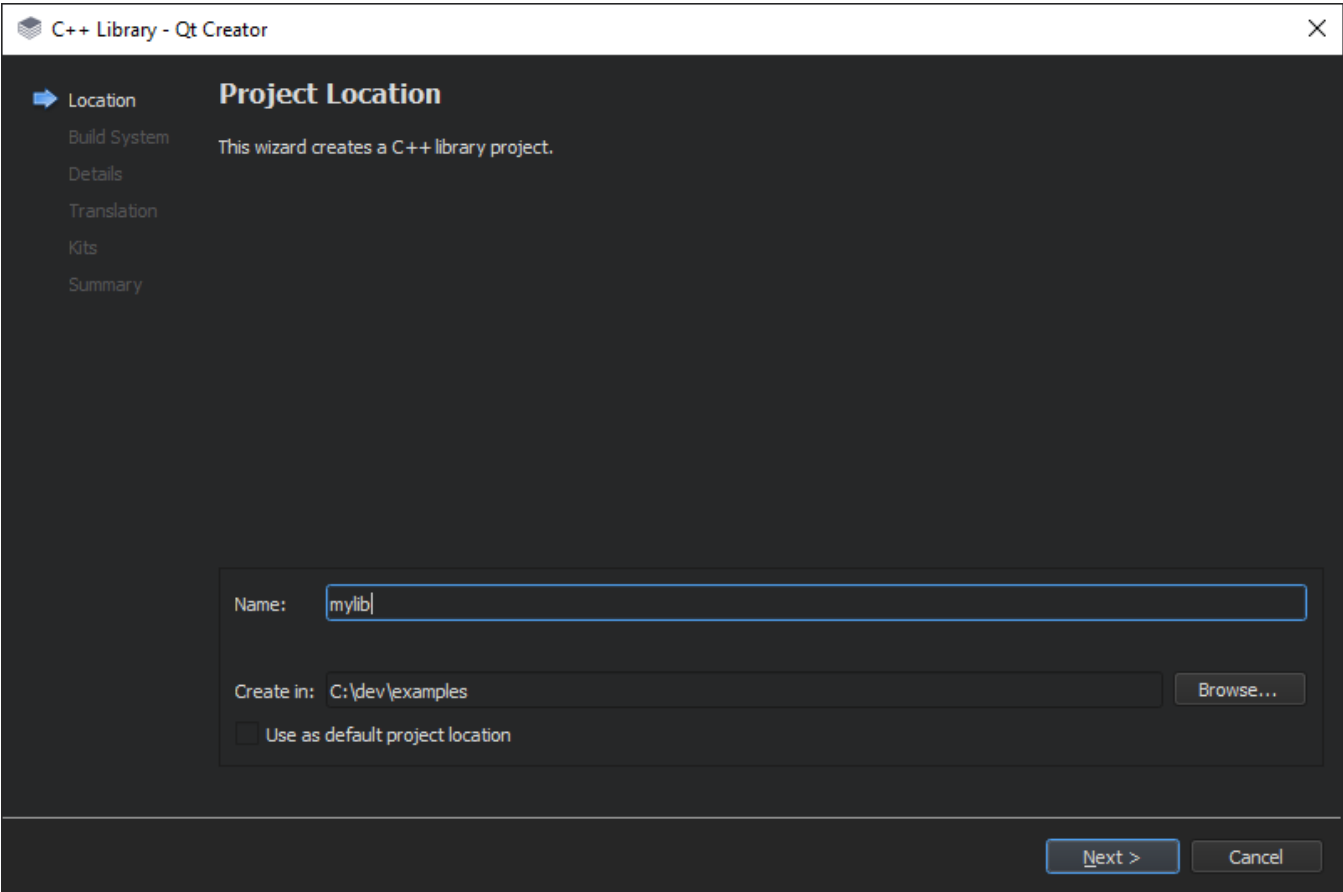
添加库

有关项目文件设置的更多信息，请参见[声明其他库](#)。

添加内部库的示例

向项目添加内部库：

- 1. 选择“**文件**>**新建项目**>**库**>**C++库**”。
- 2. 选择“选择”以打开“**项目位置**”对话框。



- 3. 在“名称”字段中，为库指定一个**名称**。例如，mylib。
- 4. 按照向导的说明进行操作，直到进入“**项目管理**”对话框。在“**作为子项目添加到项目**”列表中，选择一个项目。例如，myapp。
- 5. 在“项目”视图中，右键单击**项目名称**以打开上下文菜单，然后选择“**添加库**>**内部库**>**下一步**”。
- 6. 在“**库**”字段中，选择“mylib”，然后选择“**下一步**”。
- 7. 选择“**完成**”以将库声明添加到项目文件。

使用 CMake 时，命令将添加到 CMakeLists.txt 文件中：target_link_libraries

```
target_link_libraries(myapp PRIVATE mylib)
```

使用 qmake 时，以下库声明将添加到 .pro 文件中：

```
win32:CONFIG(release, debug|release): LIBS += -L$$OUT_PWD/../../../../projects/mylib/release/ -lmylib
else:win32:CONFIG(debug, debug|release): LIBS += -L$$OUT_PWD/../../../../projects/mylib/debug/ -lmylib
else:unix: LIBS += -L$$OUT_PWD/../../../../projects/mylib/ -lmylib
INCLUDEPATH += $$PWD/../../../../projects/mylib
DEPENDPATH += $$PWD/../../../../projects/mylib
win32:CONFIG(release, debug|release): PRE_TARGETDEPS += $$OUT_PWD/../../../../projects/mylib/release/mylib.lib
else:win32:CONFIG(debug, debug|release): PRE_TARGETDEPS += $$OUT_PWD/../../../../projects/mylib/debug/mylib.lib
```

[◀ 开放项目](#)

[添加新的自定义向导 ▶](#)

©2022 Qt有限公司 此处包含的文档贡献的版权归 他们各自的所有者。此处提供的文档根据自由软件基金会发布的[GNU 自由文档许可证版本 1.3](#)的条款进行许可。Qt和相应的徽标是Qt有限公司在芬兰和/或全球其他国家的商标。所有其他商标均为财产 其各自所有者。



The Qt Company



联系我们

公司

关于我们

投资者

编辑部

职业

办公地点

发牌

条款和条件

开源

常见问题

支持

支持服务

专业服务

合作 伙伴

训练

对于客户

支持中心

下载

Qt登录

联系我们

客户成功案例

社区

为Qt做贡献

论坛

维基

下载

市场

© 2022 Qt公司

反馈

登录