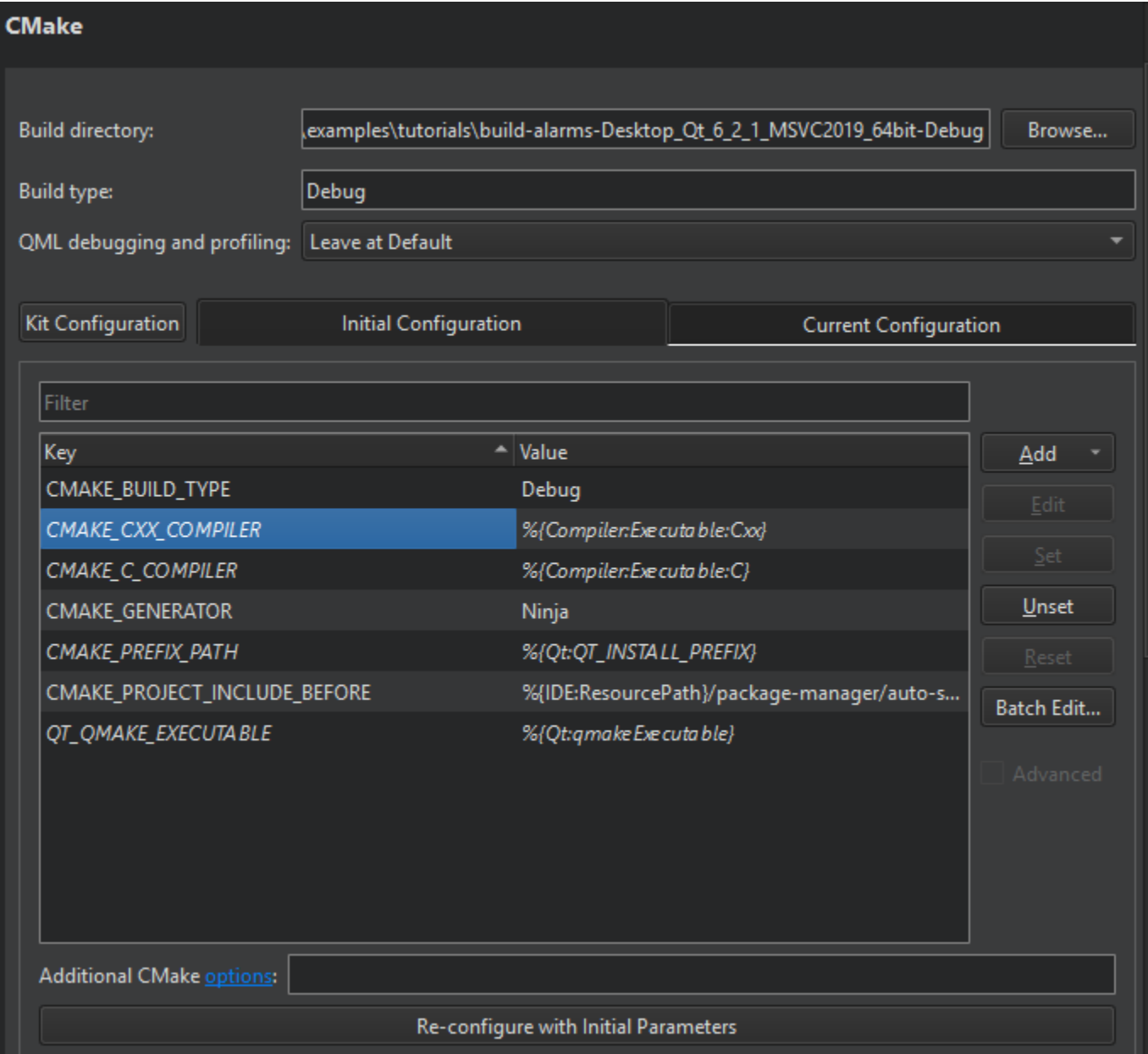


🔍 搜索

Qt创作者手册 > CMake 构建配置

# CMake 构建配置

在Qt Creator中配置大中型CMake项目可能是一个挑战，因为您需要传递给CMake才能正确配置项目的变量数量。为了简化此操作，Qt Creator会根据套件设置为您创建初始配置，并将其显示在项目构建设置的初始配置中。



**当前配置**列出了目录中 JSON 导出中的 CMake 变量。从初始配置继承的变量以斜体显示。不匹配的值以红色显示。cmake-file-api.cmake/api/v1/reply

您可以查看和编辑传递给 CMake 的变量的实际值。变量名称列在“**键**”列中，其当前值列在“**值**”列中。有关可用变量的详细信息，请在上下文菜单中选择“**帮助**”或参阅[CMake: cmake-variables \(7\)](#)。有关特定于 Qt 的变量的详细信息，请参阅[CMake 变量参考](#)。

可以指定其他 CMake 选项，例如，`...` 或在其他 CMake **选项**中。有关可用选项的更多信息，请单击字段名称中的链接或参阅[CMake: cmake \(1\)](#)。--find-debug--preset--trace-expand--warn-uninitialized

成功运行 CMake 后，您可以在**当前配置**中查看和修改当前配置。

选择“**工具包配置**”以编辑为项目选择的生成和运行工具包的 CMake 设置。

## 多配置支持

Qt Creator 支持[多配置生成器](#)，如 Xcode，Visual Studio 和 Ninja Multi-Config。这意味着您只需配置 CMake 一次，只有一个构建目录，并且可以更快地在构建类型之间切换。

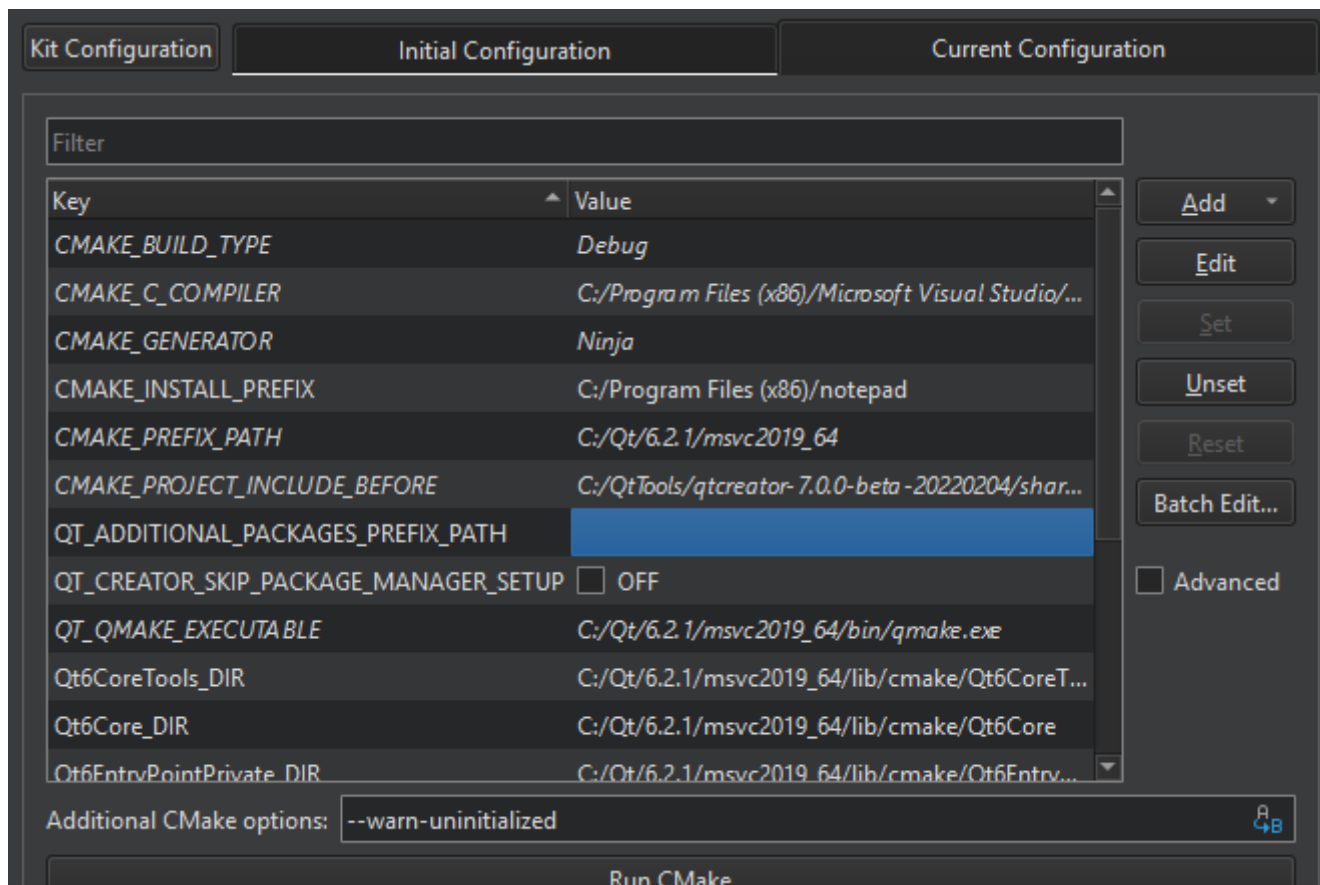
但是，这意味着 Qt Creator 不能再简单地解析第一个 CMake 文件-api JSON 导出。因此，“**生成类型**”字段的值必须与要确定的单个配置生成器（Ninja、Makefile）的变量的值匹配，以确定要使用的生成器。

CMAKE\_BUILD\_TYPE

使用 Qt 6 for iOS 进行开发时，仅支持 Xcode 生成器。

## 修改变量值

您可以查看和编辑在“**初始配置**”或“**当前配置**”中传递给 CMake 的变量的实际值。



若要添加变量，请选择“添加”，然后选择要添加的变量的类型：“布尔”、“字符串”、“目录”或“文件”。

若要更改所选变量的类型，请右键单击“键”列中的变量名称，然后在上下文菜单中选择“强制到布尔值”、“强制到文件”、“强制到目录”或“强制到字符串”。

若要将所选变量的名称或值复制到剪贴板，请在上下文菜单中选择“复制”。

若要修改变量的值，请双击它，或选择它，然后选择“编辑”。如果初始、当前和套件配置不同步，请在**初始配置**或**当前配置**的上下文菜单中选择应用**套件值**或**应用初始配置值**。

您可以一次将操作应用于多个变量。若要清除所选内容，请选择“清除所选内容”。

若要删除所选变量，请选择“取消设置”。若要撤消删除操作，请选择“设置”。

若要重置所做的所有更改，请选择“重置”。

若要修改 CMake 生成环境的环境变量值，请选择“批量编辑”。有关详细信息，请参阅[批量编辑](#)。

若要使用当前配置生成，请选择“运行 CMake”。生成时，按钮文本将更改为“停止 CMake”。选择按钮以取消当前生成。

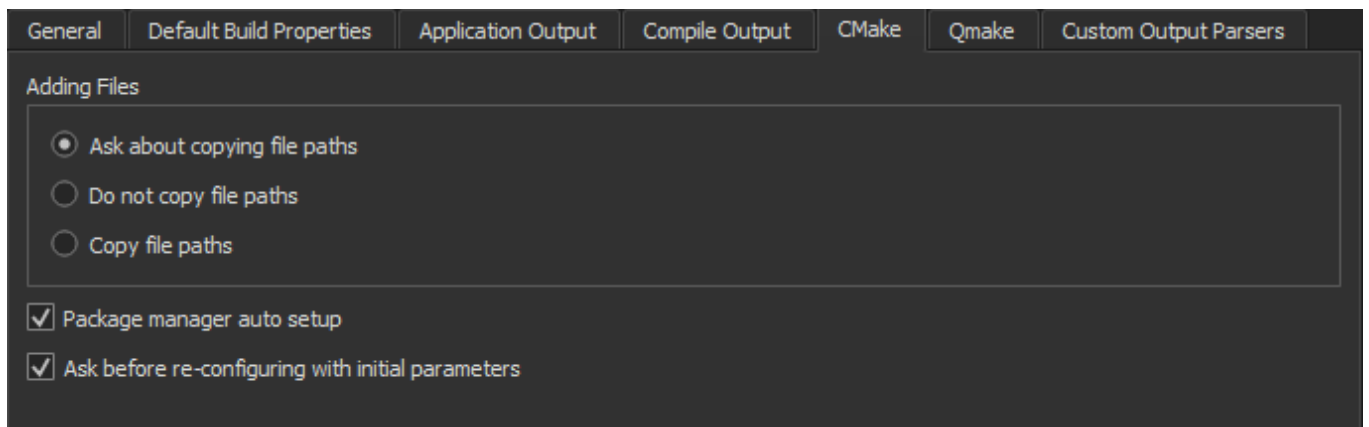
您更改的变量值通过 CMake 传递，CMake 将选项存储在 CMakeCache.txt 文件中。这意味着，如果删除生成目录，则不属于初始 CMake 配置的所有自定义变量也会被删除。-D<option>=<value>

若要使用修改的变量值重新配置项目，请选择“生成>清除 CMake 配置”，这将删除 CMakeCache.txt 文件。这使您能够执行完全重建。

## 使用初始变量重新配置

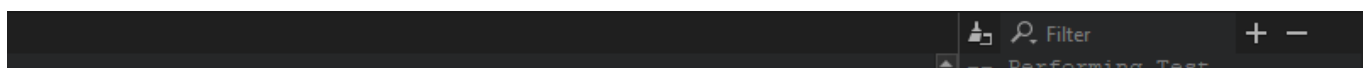
若要将 CMake 变量重置为初始变量，请选择“在初始配置中使用**初始变量重新配置**”。Qt Creator 删除当前的 CMake 配置并运行 CMake。初始配置值存储在 CMakeLists.txt.user 文件中，因此删除生成目录不会删除初始配置。

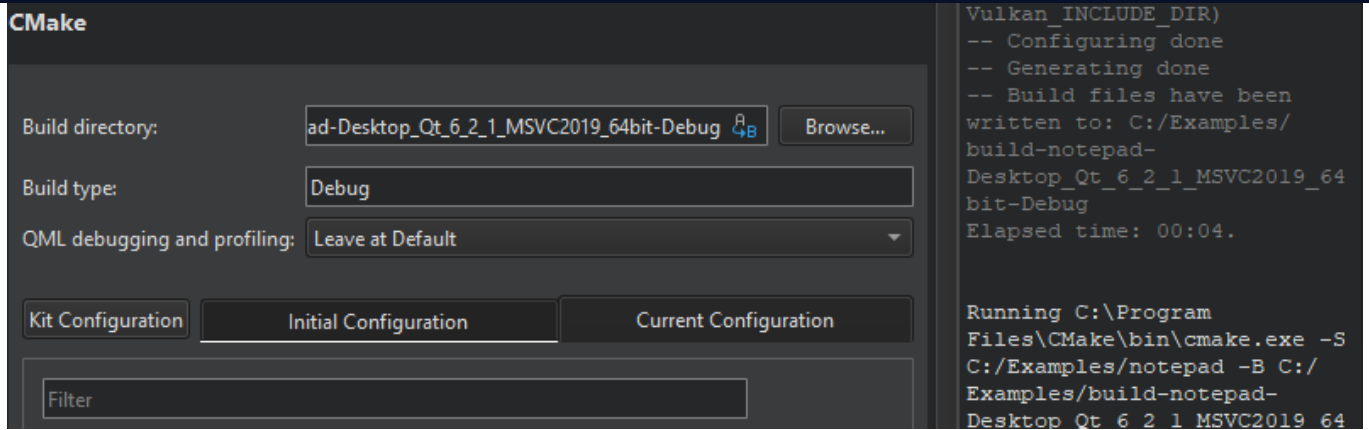
要在 Qt Creator 重置更改之前询问，请在**使用初始参数重新配置之前选择编辑**>首选项>**构建和运行**>CMake>询问。




## 查看 CMake 输出

CMake 的输出显示在“项目”模式下的“生成设置”和“运行设置”窗格旁边。





若要清除搜索结果，请选择  “(清除)”按钮。

您可以在**过滤器**字段中输入字符串以过滤输出。若要指定筛选选项，请选择该  按钮。可以使用正则表达式或区分大小写来筛选输出。选择“显示不匹配的行”以隐藏与筛选器匹配的行。

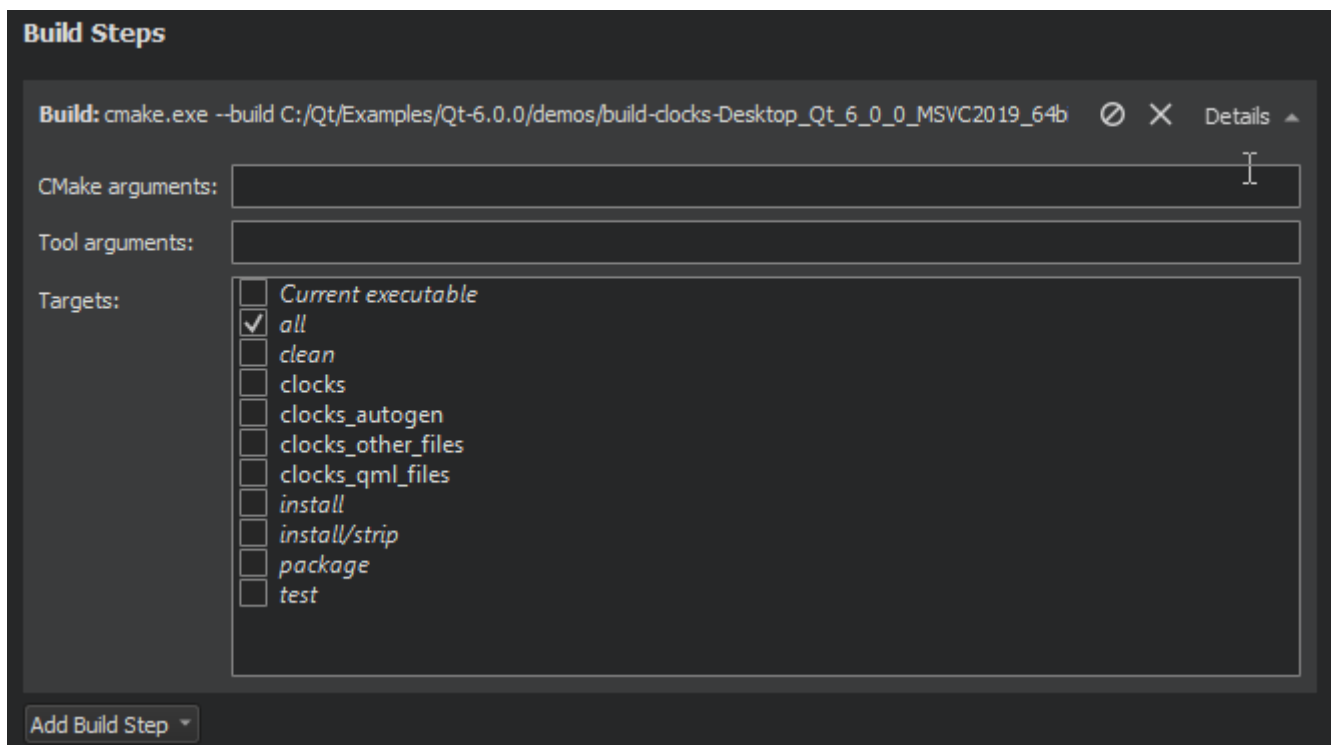
按Ctrl+F从输出中**搜索**字符串。

若要增大或减小输出文本大小，请选择“+（放大）”或“-（缩小）”，或一按Ctrl++或Ctrl+-。

## CMake 构建步骤

Qt Creator通过运行来构建CMake项目，然后运行项目配置中指定的CMake生成器：,,,或例如。CMake 生成器为Qt Creator 生成项目文件。还支持多配置生成器。cmake . --buildmakemingw32-makenmakeninja

可以在**生成步骤**中添加要传递给 CMake 的参数以及生成命令的生成器和目标。

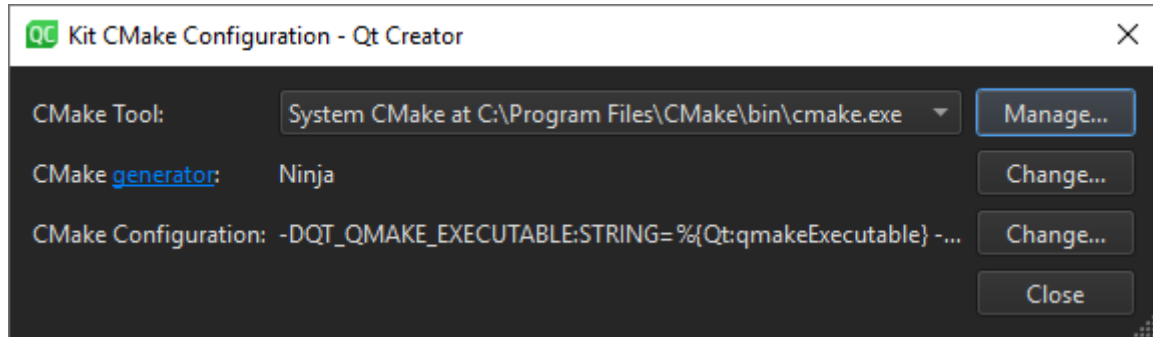


**注意：**虽然其他CMake生成器与Qt一起安装，但您通常需要自己安装Ninja。

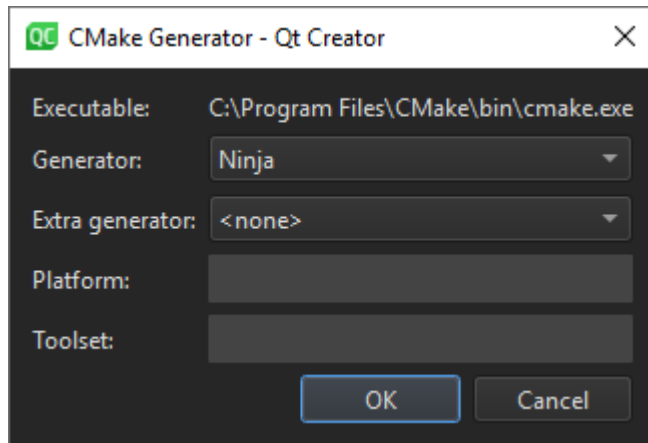
## 使用忍者作为 CMake 生成器

要将Ninja与 CMake 一起使用，您必须安装它并选择它作为构建和运行工具包中的 CMake 生成器：

1. 安装忍者。
2. 将 Ninja 可执行文件的路径添加到 PATH 系统变量的值。
3. 在“项目>生成和运行>生成>生成设置”中，选择“工具包配置”。



4. 选择“CMake生成器”字段旁边的“更改”以打开“CMake 生成器”对话框。



5. 在“生成器”中，选择“忍者”。
6. 选择“确定”以保存更改并关闭对话框。
7. 选择“关闭”以关闭“工具包 CMake 配置”对话框并返回到“生成设置”。

**注意：**若要确保旧生成项目不会妨碍更改后首次生成项目，请选择“生成>重新生成项目”。这将清理构建目录并执行新的构建。

## 将CMake与柯南一起使用

Qt Creator可以自动设置柯南包管理器以与CMake一起使用。

选择“编辑>首选项>构建和运行>CMake>包管理器自动设置”，将变量的值设置为从项目源目录中的 c、or file 安装依赖项的 CMake 脚本的路径。

CMAKE\_PROJECT\_INCLUDE\_BEFOREconanfile.txtconanfile.pyvcpkg.json

## 清正清洁步骤

Clean Steps

Build: `cmake.exe --build C:/Qt/Examples/Qt-6.0.0/demos/build-clocks-Desktop_Qt_6_0_0_MSVC2019_64b`

Details

CMake arguments:

Tool arguments:

Targets:

☐ *Current executable*

☐ *all*

☒ *clean*

☐ *clocks*

☐ *clocks\_autogen*

☐ *clocks\_other\_files*

☐ *clocks\_qml\_files*

☐ *install*

☐ *install/strip*

☐ *package*

☐ *test*

Add Clean Step

生成错误和警告将进行分析并显示在“问题”中。

◀ 指定生成设置

qmake 构建配置 ▶

©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。



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

对于客户



合作 伙伴  
训练

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