

🔍 搜索

Qt 6.4 > 使用 CMake 构建 > [CMake 变量引用](#)

# CMake 变量引用

## 模块变量

加载了带有各种变量的Qt模块。find\_package

**注意：**您很少需要直接访问这些变量。常见任务（如链接模块）应通过每个模块定义的库目标来完成。

例如，如果成功，则使以下变量可用：find\_package(Qt6 COMPONENTS Widgets)

Topics >

Qt6Widgets_COMPILE_DEFINITIONS	针对库生成时要使用的编译定义列表。
Qt6Widgets_DEFINITIONS	针对库生成时要使用的定义列表。
Qt6Widgets_EXECUTABLE_COMPILE_FLAGS	针对库生成可执行文件时使用的标志字符串。
Qt6Widgets_FOUND	描述是否成功找到模块的布尔值。
Qt6Widgets_INCLUDE_DIRS	针对库生成时要使用的包含目录列表。
Qt6Widgets_LIBRARIES	模块的导入目标的名称：Qt5::Widgets
Qt6Widgets_PRIVATE_INCLUDE_DIRS	针对库构建和使用私有 Qt API 时要使用的私有包含目录列表。
Qt6Widgets_VERSION_STRING	包含模块版本的字符串。

对于找到的所有包，这些变量的等效项可用;它们区分大小写。find\_package

## 安装变量

此外，还有一些变量与特定软件包无关，而是与Qt安装本身有关。

变量	描述
QT_DEFAULT_MAJOR_VERSION	一个整数，用于控制在混合 Qt 5 和 Qt 6 项目时转发的 Qt 版本。它需要设置为在相互调用之前任或。 Qt 5: find_package()

	5qt_qt5_6qt6_ 如果未设置，则第一个调用定义默认版本。find_package
QT_LIBINFIX	一个字符串，用于保存库名称中使用的中缀（当配置 Qt 时）。-libinfix
QT_NO_CREATE_VERSIONLESS_FUNCTIONS	隐藏以 开头的命令，仅保留以 开头的版本控制命令。qt_qt6_
QT_NO_CREATE_VERSIONLESS_TARGETS	隐藏以 开头的导入目标。相反，您需要从开始使用目标。Qt::Qt6::
QT_VISIBILITY_AVAILABLE	在 Unix 上，描述 Qt 库和插件是否编译的布尔值。这意味着仅导出选定的符号。-fvisibility=hidden

## 项目变量

这些变量会影响 Qt 提供的 CMake 命令。它们可以由项目、工具链文件或其他第三方包设置。

### Qt6：：核心

ANDROID_NDK_HOST_SYSTEM_NAME	主机系统的安卓特定架构
ANDROID_SDK_ROOT	安卓软件开发工具包的位置
QT_ANDROID_ABIS	为其构建项目包的 ABI 列表
QT_ANDROID_APPLICATION_ARGUMENTS	要传递给 Android 应用程序的参数列表
QT_ANDROID_BUILD_ALL_ABIS	支持使用自动检测的 Qt for Android SDK 列表构建多 ABI 包
QT_ANDROID_SIGN_AAB	使用指定的密钥库、别名和存储密码对 .aab 软件包进行签名
QT_ANDROID_SIGN_APK	使用指定的密钥库、别名和存储密码对软件包进行签名
QT_DEPLOY_BIN_DIR	前缀相对子目录，用于在某些目标平台上部署运行时二进制文件
QT_DEPLOY_LIB_DIR	前缀相对子目录，用于在某些目标平台上部署库
QT_DEPLOY_PLUGINS_DIR	前缀相对子目录，用于在某些目标平台上部署Qt插件
QT_DEPLOY_PREFIX	部署的基本位置
QT_DEPLOY_QML_DIR	前缀相对子目录，用于在某些目标平台上部署 QML 插件
QT_DEPLOY_SUPPORT	用于设置部署支持的文件的名称
QT_ENABLE_VERBOSE_DEPLOYMENT	启用部署工具的详细模式
QT_HOST_PATH	交叉编译时主机Qt安装的位置
QT_IOS_LAUNCH_SCREEN	所有目标使用的 iOS 启动屏幕情节提要的路径
QT_NO_COLLECT_BUILD_TREE_APK_DEPS	防止在 Android 部署期间收集项目构建的共享库目标
QT_NO_SET_XCODE_BUNDLE_IDENTIFIER	在 iOS 上禁用为目标完成期间提供回退应用捆绑包 ID
QT_NO_SET_XCODE_DEVELOPMENT_TEAM_ID	在 iOS 上禁用为目标完成期间提供回退团队 ID
QT_NO_STANDARD_PROJECT_SETUP	防止后续调用 qt_standard_project_setup () 进行任何更

Qt6: : Qml

QT_QML_OUTPUT_DIRECTORY	默认情况下将在下面创建 QML 模块的基本输出目录
-------------------------	---------------------------

Qt6: : 接口框架

< CMake 命令参考

清宗属性参考 >

©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/或其他国家/地区的商标 全球。所有其他商标均为其各自所有者的财产。



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



打印

© 2022 Qt公司

[反馈](#) [登录](#)