CMake Variable Reference

Module variables

Qt modules loaded with set various variables.`find_package`

> **Note:** You rarely need to access these variables directly. Common tasks like linking against a module should be done through the library targets each module defines.

For example, , when successful, makes the following variables available:`find_package(Qt6 COMPONENTS Widgets)`

Variable	Description
`Qt6Widgets_COMPILE_DEFINITIONS`	A list of compile definitions to use when building against the library.
`Qt6Widgets_DEFINITIONS`	A list of definitions to use when building against the library.
`Qt6Widgets_EXECUTABLE_COMPILE_FLAGS`	A string of flags to use when building executables against the library.
`Qt6Widgets_FOUND`	A boolean that describes whether the module was found successfully.
`Qt6Widgets_INCLUDE_DIRS`	A list of include directories to use when building against the library.
`Qt6Widgets_LIBRARIES`	The name of the imported target for the module: `Qt5::Widgets`
`Qt6Widgets_PRIVATE_INCLUDE_DIRS`	A list of private include directories to use when building against the library and using private Qt API.
`Qt6Widgets_VERSION_STRING`	A string containing the module's version.

For all packages found with , equivalents of these variables are available; they are case-sensitive.`find_package`

Installation variables

变量	描述
QT_DEFAULT_MAJOR_VERSION	一个整数，控制 Qt 版本，在 Qt 5 和 Qt 6 混合项目的情况下，命令转发到该版本。它需要设置为其中一个或之前相应的调用。qt_56find_package() 如果设置为 ，则以 开头的命令将调用以 开头的对应命令。 如果设置为 ，它们将调用以 开头的对应项。 5qt_qt5_6qt6_ 如果未设置，则第一个调用将定义默认版本。 find_package
QT_LIBINFIX	一个字符串，用于保存库名称中使用的中缀，当 Qt 配置了 。-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，您需要使用以 开头的目标。Qt::Qt6::
QT_VISIBILITY_AVAILABLE	在 Unix 上，一个布尔值，描述 Qt 库和插件是否使用 编译。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软件包设置。

Qt6：：核心

ANDROID_NDK_HOST_SYSTEM_NAME	特定于安卓系统的主机系统架构
ANDROID_SDK_ROOT	安卓开发工具包的位置
QT_ANDROID_ABIS	构建项目文件包所依据的 ABI 列表
QT_ANDROID_APPLICATION_ARGUMENTS	要传递给安卓应用程序的参数列表
QT_ANDROID_BUILD_ALL_ABIS	支持使用自动检测的适用于安卓的 Qt 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 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	用于在某些目标平台上部署运行时二进制文件的前缀相对子目录
QT_DEPLOY_LIB_DIR	用于在某些目标平台上部署库的前缀相对子目录
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	要包含的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_NO_SET_XCODE_BUNDLE_IDENTIFIER	禁止在 iOS 上完成目标期间提供回退应用捆绑 ID
QT_NO_SET_XCODE_DEVELOPMENT_TEAM_ID	禁止在 iOS 上完成目标期间提供回退团队 ID
QT_NO_STANDARD_PROJECT_SETUP	防止对 qt_standard_project_setup（） 的后续调用进行任何更改
QT_PATH_ANDROID_ABI_<阿比>	用于为相应的 ABI 指定安卓版 Qt 路径的变量集

Qt6：

QT_QML_OUTPUT_DIRECTORY	基本输出目录，默认情况下将在其下创建 QML 模块

Qt6：：接口框架

‹ 命令参考 断续器属性参考 ›

社区

为Qt做贡献
论坛
维基
下载
市场

反馈 登录