Search

CMake Variable Reference

Module variables

Qt modules loaded with `find_package` set various variables.

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

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 `find_package`, equivalents of these variables are available; they are case-sensitive.

Installation variables

Qt DOCUMENTATION

Variable	Description
QT_DEFAULT_MAJOR_VERSION	An integer that controls the Qt version that `qt_` commands forward to in case of mixed Qt 5 and Qt 6 projects. It needs to be set to either 5 or 6 before the respective `find_package()` calls. If set to 5, commands starting with `qt_` will call their counterpart starting with `qt5_`. If set to 6, they will call their counterpart starting with `qt6_`. If not set, the first `find_package` call defines the default version.
QT_LIBINFIX	A string that holds the infix used in library names, when Qt is configured with `-libinfix`.
QT_NO_CREATE_VERSIONLESS_FUNCTIONS	Hides commands that start with `qt_`, leaving only the versioned ones starting with `qt6_`.
QT_NO_CREATE_VERSIONLESS_TARGETS	Hides the imported targets starting with `Qt::`. Instead, you need to use the targets starting with `Qt6::`.
QT_VISIBILITY_AVAILABLE	On Unix, a boolean that describes whether Qt libraries and plugins were compiled with `-fvisibility=hidden`. This means that only selected symbols are exported.

Topics

Project variables

These variables can influence CMake commands provided by Qt. They may be set by the project, a toolchain file or other third-party packages.

Qt6::Core

ANDROID_NDK_HOST_SYSTEM_NAME	Android-specific architecture of the host system
ANDROID_SDK_ROOT	Location of the Android SDK
QT_ANDROID_ABIS	List of ABIs that the project packages are built for
QT_ANDROID_APPLICATION_ARGUMENTS	List of arguments to pass to Android applications
QT_ANDROID_BUILD_ALL_ABIS	Enables building multi-ABI packages using the autodetected Qt for Android SDK list
QT_ANDROID_SIGN_AAB	Sign the .aab package with the specified keystore, alias and store password
QT_ANDROID_SIGN_APK	Sign the package with the specified keystore, alias and store password
QT_DEPLOY_BIN_DIR	Prefix-relative subdirectory for deploying runtime binaries on some target platforms
QT_DEPLOY_LIB_DIR	Prefix-relative subdirectory for deploying libraries on some target platforms
QT_DEPLOY_PLUGINS_DIR	Prefix-relative subdirectory for deploying Qt plugins on some target platforms
QT_DEPLOY_PREFIX	Base location for a deployment

Qt DOCUMENTATION

QT_DEPLOY_SUPPORT	Name of the file to include for setting up deployment support
QT_ENABLE_VERBOSE_DEPLOYMENT	Enables verbose mode of deployment tools
QT_HOST_PATH	Location of the host Qt installation when cross-compiling
QT_IOS_LAUNCH_SCREEN	Path to iOS launch screen storyboard used by all targets
QT_NO_COLLECT_BUILD_TREE_APK_DEPS	Prevents collecting of project-built shared library targets during Android deployment
QT_NO_SET_XCODE_BUNDLE_IDENTIFIER	Disables providing a fallback app bundle ID during target finalization on iOS
QT_NO_SET_XCODE_DEVELOPMENT_TEAM_ID	Disables providing a fallback team ID during target finalization on iOS
QT_NO_STANDARD_PROJECT_SETUP	Prevents subsequent calls to qt_standard_project_setup() from making any changes
QT_PATH_ANDROID_ABI_<ABI>	Set of variables to specify the path to Qt for Android for the corresponding ABI

Qt6::Qml

QT_QML_OUTPUT_DIRECTORY	Base output directory below which QML modules will be created by default

Qt6::InterfaceFramework

‹ CMake Command Reference

CMake Property Reference ›

Support

Support Services
Professional Services
Partners
Training

For Customers

Support Center
Downloads
Qt Login
Contact Us
Customer Success

Community

Contribute to Qt
Forum
Wiki
Downloads
Marketplace

Feedback Sign In