Supporting Derived Projects

Contents

Example Manifest Files

QDoc generates XML files that contain information about all documented examples and demos. These files, named `examples-manifest.xml` and `demos-manifest.xml`, are used by Qt Creator to present a list of examples in its welcome screen and to link to their documentation.

Manifest XML Structure

A manifest file has the following structure:

```
<?xml version="1.0" encoding="UTF-8"?>
<instructionals module="QtGui">
    <examples>
        <example
            name="Analog Clock Window Example"
            docUrl="qthelp://org.qt-project.qtgui.502/qtgui/analogclock.html"
            projectPath="gui/analogclock/analogclock.pro"
            imageUrl="qthelp://org.qt-project.qtgui.502/qtgui/images/analogclock-window-example.png">
            <description><![CDATA[The Analog Clock Window example shows how
                to draw the contents of a custom window.]]></description>
            <tags>analog,clock,window</tags>
            <fileToOpen>gui/analogclock/main.cpp</fileToOpen>
        </example>
        ...
    </examples>
</instructionals>
```

Each `<example>` element contains information about a name, description, the location of the project file and documentation, as well as a list of tags associated with the example.

Manifest Meta Content

It is possible to augment the manifest files with additional meta-content - that is, extra attributes and tags for selected examples, using the `manifestmeta` configuration command.

One use case for meta-content is highlighting a number of prominent examples. Another is improving search functionality by adding relevant keywords as tags for a certain category of examples.

The examples for which meta-content is applied to is specified using one or more filters. Matching examples to filters is done based on names, with each example name prefixed with a module name and a slash. Simple wildcard matching is supported; by using `*` at the end it's possible to match multiple examples with a single string.

Example:

```
manifestmeta.filters                = highlighted sql webkit global

manifestmeta.highlighted.names      = "QtGui/Analog Clock Window Example" \
                                      "QtWidgets/Analog Clock Example"
manifestmeta.highlighted.attributes = isHighlighted:true
manifestmeta.sql.names              = "QtSql/*"
manifestmeta.sql.tags               = database,sql

manifestmeta.webkit.names           = "QtWebKitExamples/*"
manifestmeta.webkit.tags            = webkit

manifestmeta.global.names           = *
manifestmeta.global.tags            = qt5
```

Above, an `isHighlighted` attribute is added to two examples. If the attribute value is omitted, QDoc uses the string `true` by default. Extra tags are added for Qt WebKit and Qt SQL examples, and another tag is applied to all examples by using just `*` as the match string.

Supporting Derived Projects