



QT Project File (.pro)



핵심 정리

- **.pro 파일**

- ⇒ QT project 를 관리하는 파일
- ⇒ 소스(헤더) 파일을 추가 할 때 마다 이 파일에 항목 추가
- ⇒ 주석은 # 을 사용



핵심 정리

● QT Modules

⇒ 다양한 종류의 클래스를 “**모듈**” 단위로 분리해서 관리

⇒ 특정 모듈을 사용하려면 .pro 파일에 “**QT += module이름**” 으로 추가해야 함.

.pro 파일에 추가하지
않아도 디폴트로 포함 됨.



Qt Core

Qt GUI

Qt D-Bus

Qt Network

Qt QML

Qt Quick

Qt Quick Controls

Qt Quick Dialogs

Qt Quick Layouts

Qt Quick Test

Qt Test

Qt Widgets



QDebug()



핵심 정리

예제 Project

Project Name	QDEBUG
Project Type	Empty qmake Project
Source	main.cpp



핵심 정리

● QDebug

- ⇒ 화면 출력을 위해 QT 가 제공하는 클래스
- ⇒ <QDebug> 헤더
- ⇒ 다양한 QT 클래스의 객체를 편리하게 출력 가능.
- ⇒ qDebug() 함수를 사용하면 “**화면 출력을 위한 QDebug 타입의 디폴트 객체**” 를 얻을 수 있다

QDebug

클래스

일반적으로 이 클래스를 직접 사용하기 보다는 “qDebug()” 함수 사용

qDebug()

함수, 화면 출력을 위한 QDebug 타입의 객체를 반환

● qDebug() 사용법

- ⇒ printf 스타일
- ⇒ cout 스타일



event loop

QCoreApplication



핵심 정리

예제 Project

Project Name	QCoreApplication
Project Type	Empty qmake Project
Source	main.cpp



● QTimer

- ⇒ 시간 간격에 따라 반복되거나, 특정 시간에 이벤트가 발생하도록 하는 클래스
- ⇒ 타이머 이벤트를 처리하려면 프로그램이 종료 되면 안된다.

● event loop

- ⇒ 프로그램을 종료하지 않고 발생하는 이벤트를 처리하기 위한 사용
- ⇒ “**Timer, Network, GUI**” 관련 코드를 작성하는 경우 반드시 이벤트 루프가 필요 하다.



핵심 정리

- event loop 제공하기 위한 QT 클래스

QObject

QT의 최상위 클래스

대부분 QT 클래스가
가져야 하는 기본 기능 제공

non-GUI Application

QCoreApplication

GUI를 사용하지 않은 프로그램을
위한 event loop 제공

QGuiApplication

GUI를 사용 하는 프로그램을
위한 event loop 제공

GUI Application

QApplication

QWidget 을 **사용** 하는 프로그램을
위한 event loop 제공



핵심 정리

- event loop 를 가진 non-GUI 프로그램의 전형적인 main 함수

```
#include <QCoreApplication>

int main(int argc, char** argv)
{
    QCoreApplication app(argc, argv);

    // .....

    return app.exec();
}
```



Help

QT Documentation



핵심 정리

예제 Project

Project Name

Documentation

Project Type

Empty qmake Project

Source

main.cpp



핵심 정리

- 특정 클래스에 대한 자세한 정보를 알고 싶다.
- QT 도움말(reference)
 - ① **QT Creator** 의 내장 도움말
 - ② **qt.io** 사이트에 접속

The screenshot shows the Qt Group website. The top navigation bar includes 'Qt Group', 'Products', 'Solutions', and 'Resources'. A red dashed box labeled '1' highlights the 'Resources' menu. The 'Resources' dropdown menu is open, showing sections: 'Development Framework and Tools', 'THE LATEST Qt RESOURCES' (with links to Qt Blog, Live Webinars and Events, Qt Customer cases, Qt Demos, and All Qt resources), 'LEARN Qt TOOLS' (with links to Qt Academy and Get Qt Educational license), and 'CONNECT WITH Qt COMMUNITY'. A red dashed box labeled '2' highlights the 'Qt Documentation' link under the 'CONNECT WITH Qt COMMUNITY' section. The background of the website features a large heading 'The Future of Digital Experience' and a subheading 'All the tools you need for creating software for embedded devices, from planning and development, testing, and future-proofing'.

Qt Group Products Solutions **1** Resources

The Future of Digital Experience

All the tools you need for creating software for embedded devices, from planning and development, testing, and future-proofing

Development Framework and Tools

THE LATEST Qt RESOURCES

- > Qt Blog
- > Live Webinars and Events
- > Qt Customer cases
- > Qt Demos
- > All Qt resources

LEARN Qt TOOLS

- > Qt Academy
- > Get Qt Educational license

CONNECT WITH Qt COMMUNITY

2 Qt Documentation

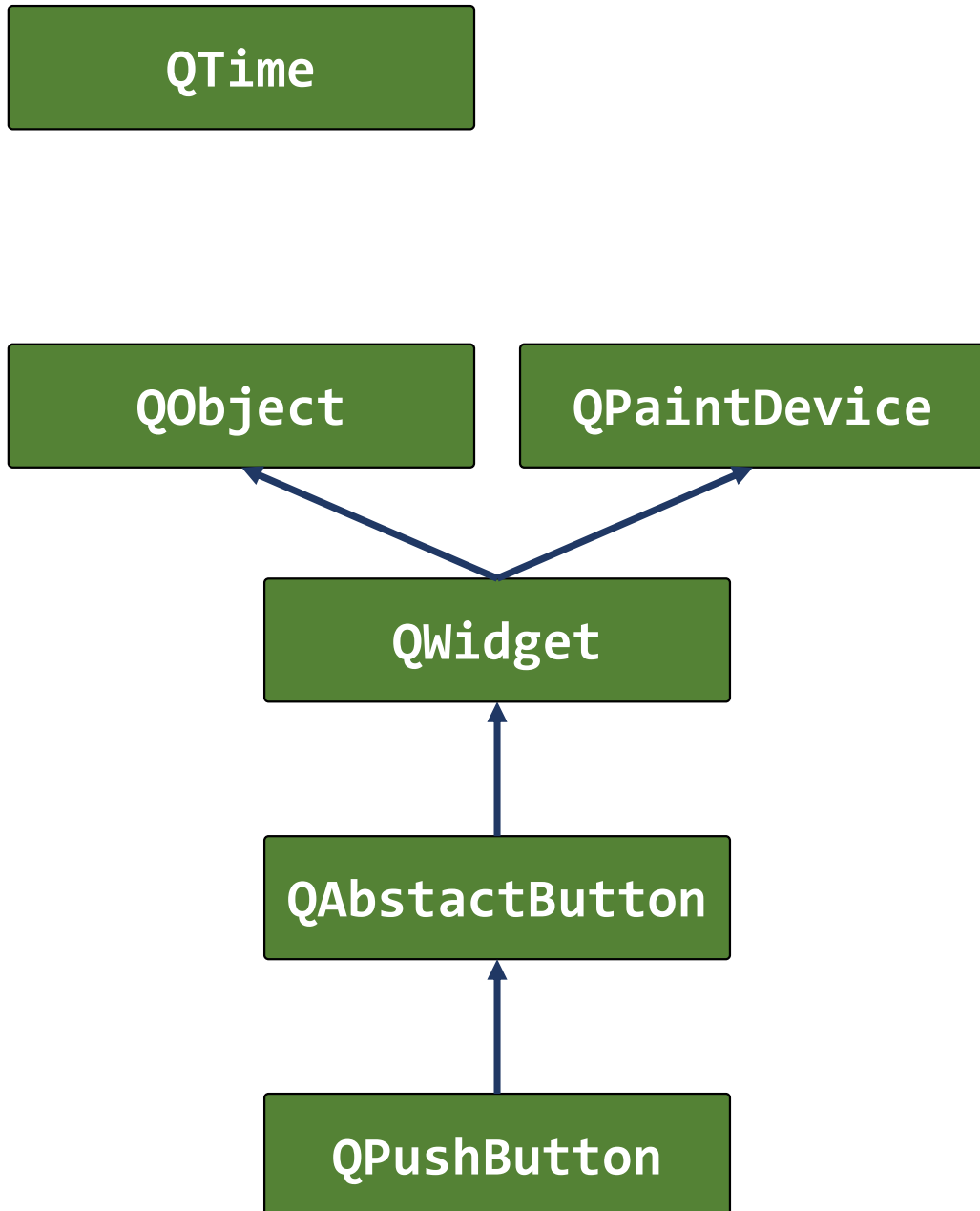
- > Qt Forum
- > Qt Wiki
- > Contribute to Qt

<https://www.qt.io/resources/qt?content-type=Demo>



핵심 정리

- 검색 창에서 클래스 이름 검색
- 클래스 계층도를 항상 살펴 볼 것





핵심 정리

- QT 소스 코드 확인

- ⇒ 클래스 이름에서 마우스 오른쪽 버튼 누른 후에
“**Follow Symbol Under Cursor(F2)**” 메뉴 선택
(위에서 2번째 메뉴)
- ⇒ “**Show Preprocessed Source**” 를 선택하면
매크로가 제거된 후의 코드도 확인 가능