**Getting started with Qt**

 Download Qt (Open Source LGPL license) from <http://qt-project.org/downloads>. The download recommended by the website is the online installer, but you should not download that as it is buggy. Click "View All Downloads" at the end of the page and choose a suitable option for Qt 5.7.1. For Windows users, if you have other compiler installed already like Visual Studio or MinGW, you should download the one that matches your system. If not, it is advisable to download the MinGW version of Qt (the 32-bit version works fine on 64-bit machines). If that doesn't work too, you need to download compiler to make Qt creator compile your code. Finally install Qt by following the instructions in the dialog boxes that appear. You can find help with installation on Youtube [here](https://www.youtube.com/watch?v=eS7ank-qFjg&index=2&list=PLS1QulWo1RIZiBcTr5urECberTITj7gjA).

**Using Qt’s editor and compiler:**

1. Run “Qt Creator”
	1. By double clicking the ImgFilter.pro file in the starter kit folder or by selecting “Open file or project” from File menu in Qt Creator window and then the file “ImgFilter.pro”.
	2. Click “Configure Project.”
	3. Click on the “Sources” folder. You should see a file called Project1.cpp. This is the only file you should need to edit.
	4. To compile and run, click on the green play button at the bottom left of window. The application window will open on the screen. Select the desired image from the "Data" folder using the "Open image" button and then click the other buttons to perform the required operations. Select "Save image" and enter "<filename>.png" in the Filename text box to save the output images.
	5. You can select between debug and release builds above the green play button.

**Using Visual Studio’s editor and compiler:**

1. Run “Visual Studio” (2010 preferable)
	1. Select “Open Project/Solution” from File menu.
	2. Select the file “ImgFilter.vcproj”
	3. Right click on the ImgFilter project and select “Properties”
	4. Select “Configuration Properties -> C/C++ -> General
	5. Update the directories under “Additional Include Directories” with “Qt” in the name according to where you installed Qt.
	6. In the same window select “Configuration Properties -> Linker”.





* 1. Update the directory in “Additional Library Directories” according to where you installed Qt and add the libraries.
	2. Compile and Run.
	3. If it complains about not finding Qt’s dlls when running, you should update your system’s “path” variable to include Qt’s “bin” directory. To update the “path” variable, type “environment” into the window search box and select “Edit the system’s environment variables”. Press the “Environment Variable…” button and find “path” in the second list box.
	4. Qt also has a VS add-in that is nice for editing UIs. This isn’t necessary for the assignment, but you can find it at the same download page. [**Visual Studio Add-in 1.2.2 for Qt5**](http://download.qt-project.org/official_releases/vsaddin/qt-vs-addin-1.2.2-opensource.exe) is preferable.
	5. You shouldn’t have to do this – but just in case. If you’d like to convert a Qt project to a Visual Studio project use the following command “qmake -tp vc test.pro”. This will create a file test.vcproj.
	6. If it complains about MOC/UIC not found, make sure the Qt Project Settings look like this and the modules should be selected like this:

 