# Task 4 – Qt Exercise Explanation

## Task:

Explain, providing a source snippet, how you would modify Qt Text Edit tutorial to load and display the files from task 1.

## Answer:

Completing the 5-step tutorial the only extension of the code that is required is the following:

Original Code:



Modified Code:



In the open() function the static function getOpenFileName() is used to get the path of the file the user selects. File filters are set to text (.txt file extensions), C++ (.cpp file extensions) and header files (.h file extensions) originally, but this is extended to include Data files (.dat file extensions).

# Task 6 – Qwt Exercise Explanation

## Task:

Explain, providing a source snippet, how you would modify Qwt simple plot example to plot the amplitude curve calculated in task 1. Note: the x-axis is sample number; y-axis is amplitude.

## Answer:

The points are set with the following code:



Where the QPointF() is used to set points on the graph and each is inputted into the QPolygonF() variable points.

Making use of the code used in task 1 to read and parse the data from a file, the output from task 1 is used as an input and each amplitude number in column 1 is stored in an array of doubles called amplitudes:

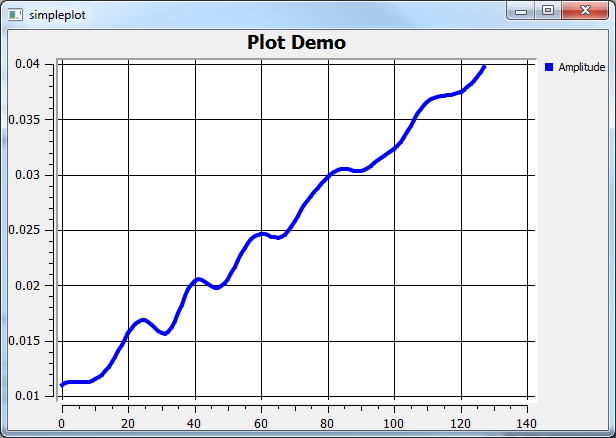


Finally the code originally used to plot points on the graph is replaced with the following:



This iterates through the amplitudes array inputting a QPointF for each amplitude number (y-axis) alongside the float cast of the current counter (x-axis) to the graph.

This produces the following output (with tweaked size-steps on the x-axis):



# Task 6 – Qwt Exercise Explanation (Update)

## Task:

Explain, providing a source snippet, how you would modify Qwt simple plot example to plot the amplitude curve calculated in task 1. Note: the x-axis is sample number; y-axis is amplitude.

## Answer:

The curves in the original tutorial are set with the following code:



Where setData() is used to set the points on the curve. However, in the tutorial the points are generated from a sin wave and not set individually thus the following code has been modified from the reworked task 1:



This code sets up the curve in a similar manner to the tutorial, but then makes use of the file read and extraction code from task 1 to extract amplitude numbers from the file and push them into the QPolygonF variable to hold points. The counter i is used to set the sample number variable on the x-axis, and the amplitude variable pulled from the input file is set to the y-axis. The points variable is then input to the cAmplitude curve.

Due to issues with building Qwt 5.2.1, this solution could not be tested and thus a output graph could not be generated.