

C++ Advanced (with QT)

This Advanced C++ with QT Training course will teach you how to create hybrid desktop/Internet applications, threading, rich text handling, Qt's graphics/view architecture, and Qt's model/view architecture and many other topics are covered. Besides each chapter's main subject, lots of other classes, methods, and techniques are used wherever they make sense, to show as many Qt features as possible.

Prerequisites

You should be solidly at the [C++ Beginner](#) level before doing this course.

Further Training

[HTML5 / CSS3 Responsive Web Design](#) (if you want to learn more about creating responsive front-ends).

How about JavaScript, Angular, Ionic ?

Course Material

Included

Course Contents

Hybrid Desktop/Internet Applications

- Internet-Aware Widgets
- Using WebKit
- A Generic Web Browser Window Component
- Creating Web Site-Specific Applications
- Embedding Qt Widgets in Web Pages

Audio and Video

- Using QSound and QMovie
- The Phonon Multimedia Framework

- Playing Music
- Playing Videos

Model/View Table Models

- Qt's Model/View Architecture
- Using QStandardItemModels for Tables
- Changing a Table Model through the User Interface
- A QStandardItemModel Subclass for Tables
- A QSortFilterProxyModel to Filter Out Duplicate Rows
- A QSortFilterProxyModel to Filter In Wanted Rows
- Creating Custom Table Models
- Changing a Table Model through the User Interface
- A Custom QAbstractTableModel Subclass for Tables
- The QAbstractItemModel API Methods for Tables
- Methods to Support Saving and Loading Table Items

Model/View Tree Models

- Using QStandardItemModels for Trees
- Changing a Tree Model through the User Interface
- A QStandardItem Subclass for Tree Items
- A QStandardItemModel Subclass for Trees
- Creating Custom Tree Models
- Changing a Tree Model through the User Interface
- A Custom Item Class for Tree Items
- A Custom QAbstractItemModel Subclass for Trees
- The QAbstractItemModel API for Trees
- The QAbstractItemModel API for Drag and Drop
- Methods for Saving and Loading Tree Items

Model/View Delegates

- Datatype-Specific Editors
- Datatype-Specific Delegates
- A Read-Only Column or Row Delegate
- An Editable Column or Row Delegate
- Model-Specific Delegates

Model/View Views

- QAbstractItemView Subclasses
- Model-Specific Visualizing Views
- The Visualizer Widget
- The Visualizer's Aggregated Header Widget
- The Visualizer's Aggregated View Widget

Threading with QtConcurrent

- Executing Functions in Threads
- Using QtConcurrent::run()
- Using QRunnable
- Filtering and Mapping in Threads
- Using QtConcurrent to Filter
- Using QtConcurrent to Filter and Reduce
- Using QtConcurrent to Map

Threading with QThread

- Processing Independent Items
- Processing Shared Items

Creating Rich Text Editors

- Introducing QTextDocument
- Creating Custom Text Editors
- Completion for Line Edits and Comboboxes
- Completion and Syntax Highlighting for Text Editors
- Completion for Multi-line Editors
- Syntax Highlighting
- A Rich Text Single Line Editor
- Multi-line Rich Text Editing

Creating Rich Text Documents

- Exported QTextDocument File Quality
- Creating QTextDocuments
- Creating QTextDocuments with HTML
- Creating QTextDocuments with QTextCursor

- Exporting and Printing Documents
- Exporting QTextDocuments
- Exporting in PDF and PostScript Format
- Exporting in Open Document Format
- Exporting in HTML Format
- Exporting in SVG Format
- Exporting in Pixmap Formats
- Printing and Previewing QTextDocuments
- Painting Pages
- Painting PDF or PostScript
- Painting SVG
- Painting Pixmap

Creating Graphics/View Windows

- The Graphics/View Architecture
- Graphics/View Widgets and Layouts
- Introducing Graphics Items

Creating Graphics/View Scenes

- Scenes, Items, and Actions
- Creating the Main Window
- Saving, Loading, Printing, and Exporting Scenes
- Saving Scenes
- Loading Scenes
- Printing and Exporting Scenes
- Manipulating Graphics Items
- Adding Items
- Copying, Cutting, and Pasting Items
- Manipulating Selected Items
- Showing and Hiding the Guideline Grid
- Keeping the User Interface Up to Date
- Enhancing QGraphicsView
- Creating a Dock Widget Toolbox
- Creating Custom Graphics Items
- Enhancing QGraphicsTextItem
- Graphics Item Transformations

- Enhancing an Existing Graphics Item
- Creating a Custom Graphics Item from Scratch

The Animation and State Machine Frameworks

- Introducing the Animation Framework
- Introducing the State Machine Framework
- Combining Animations and State Machines

PROJECT

—

Duration and pricing

In [Price Group A](#)

Certificate

1. Upon completion of this course we will issue you with attendance certificate to certify your attendance and / or completion of the prescribed minimum examples.
2. You may sit for our competency assessment test and on passing you will obtain our competency certificate.
3. Our competency assessment can be booked and taken by someone who has not attended the course at a cost of R2950.

Bookings

You can download the course registration form on our home page or by clicking [here](#)

Brochure

You may download a pdf copy of this page by clicking on the pdf icon at the top of the page.

Questions

Please [email us](#)

Schedule

On the calendar below. If your browser doesn't display the calendar below, please click on [this link](#) or try using [Google Chrome](#), alternatively please enquire via our [Contact Us](#) page.