

OPEN SOURCE SOFTWARE DEVELOPMENT SERIES

An Introduction to Design Patterns in C++ with Qt[™]

Second Edition



Alan Ezust • Paul Ezust

Foreword by Lars Knoll, Director Qt Research and Development

An Introduction to Design Patterns in C++ with QtTM, 2nd Edition

Introduction to Design Patterns in C++ with Qt

Table of Contents

Cover

Half Title

Title Page

Copyright Page

Contents

Foreword

Preface

Preface to the Second Edition

Acknowledgments

About the Authors

Part I: Design Patterns and Qt 4

Chapter 1: C++ Introduction

1.1 Overview of C++

1.2 A Brief History of C++

1.3 C++ First Example

1.4 Standard Input and Output

1.5 Introduction to Functions

1.6 qmake, Project Files, and Makefile

1.7 Getting Help Online

1.8 Strings

1.9 Streams

Table of Contents

- 1.10 File Streams
- 1.11 Qt Dialogs for User Input/Output
- 1.12 Identifiers, Types, and Literals
- 1.13 C++ Simple Types
- 1.14 The Keyword `const`
- 1.15 Pointers and Memory Access
- 1.16 Reference Variables
- 1.17 `const*` and `*const`
- 1.18 Review Questions

Chapter 2: Top of the class

- 2.1 First, There Was `struct`
- 2.2 Class Definitions
- 2.3 Member Access Specifiers
- 2.4 Encapsulation
- 2.5 Introduction to UML
- 2.6 Friends of a Class
- 2.7 Constructors
- 2.8 Destructors
- 2.9 The Keyword `static`
- 2.10 Class Declarations and Definitions
- 2.11 Copy Constructors and Assignment Operators
- 2.12 Conversions
- 2.13 `const` Member Functions
- 2.14 Subobjects
- 2.15 Exercise: Classes
- 2.16 Review Questions

Chapter 3: Introduction to Qt

- 3.1 Style Guidelines, Naming Conventions
- 3.2 The Qt Core Module

Table of Contents

3.3 QtCreatorAn IDE for Qt Programming

3.4 Exercises: Introduction to Qt

3.5 Review Questions

Chapter 4: Lists

4.1 Introduction to Containers

4.2 Iterators

4.3 Relationships

4.4 Exercise: Relationships

4.5 Review Questions

Chapter 5: Functions

5.1 Overloading Functions

5.2 Optional Arguments

5.3 Operator Overloading

5.4 Parameter Passing by Value

5.5 Parameter Passing by Reference

5.6 References to const

5.7 Function Return Values

5.8 Returning References from Functions

5.9 Overloading on const

5.10 inline Functions

5.11 Functions with Variable-Length Argument Lists

5.12 Exercise: Encryption

5.13 Review Questions

Chapter 6: Inheritance and Polymorphism

6.1 Simple Derivation

6.2 Derivation with Polymorphism

6.3 Derivation from an Abstract Base Class

6.4 Inheritance Design

6.5 Overloading, Hiding, and Overriding

Table of Contents

6.6 Constructors, Destructors, and Copy Assignment Operators

6.7 Processing Command-Line Arguments

6.8 Containers

6.9 Managed Containers, Composites, and Aggregates

6.10 Containers of Pointers

6.11 Review Questions

Chapter 7: Libraries and Design Patterns

7.1 Building and Reusing Libraries

7.2 Exercise: Installing Libraries

7.3 Frameworks and Components

7.4 Design Patterns

7.5 Review Questions

Chapter 8: QObject, QApplication, Signals, and Slots

8.1 Values and Objects

8.2 Composite Pattern: Parents and Children

8.3 QApplication and the Event Loop

8.4 Q_OBJECT and moc: A checklist

8.5 Signals and Slots

8.6 QObject Lifecycle

8.7 QTestLib

8.8 Exercises: QObject, QApplication, Signals, and Slots

8.9 Review Questions

Chapter 9: Widgets and Designer

9.1 Widget Categories

9.2 Designer Introduction

9.3 Dialogs

9.4 Form Layout

9.5 Icons, Images, and Resources

9.6 Layout of Widgets

Table of Contents

- 9.7 Designer Integration with Code
- 9.8 Exercise: Input Forms
- 9.9 The Event Loop: Revisited
- 9.10 Paint Events, Drawing Images
- 9.11 Review Questions

Chapter 10: Main Windows and Actions

- 10.1 QActions, QMenus, and QMenuBar
- 10.2 Regions and QDockWidgets
- 10.3 QSettings: Saving and Restoring Application State
- 10.4 Clipboard and Data Transfer Operations
- 10.5 The Command Pattern
- 10.6 tr() and Internationalization
- 10.7 Exercises: Main Windows and Actions
- 10.8 Review Questions

Chapter 11: Generics and Containers

- 11.1 Generics and Templates
- 11.2 Generics, Algorithms, and Operators
- 11.3 Sorted Map Example
- 11.4 Function Pointers and Functors
- 11.5 Flyweight Pattern: Implicitly Shared Classes
- 11.6 Exercise: Generics
- 11.7 Review Questions

Chapter 12: Meta Objects, Properties, and Reflective Programming

- 12.1 QMetaObjectThe MetaObject Pattern
- 12.2 Type Identification and qobject_cast
- 12.3 Q_PROPERTY MacroDescribing QObject Properties
- 12.4 QVariant Class: Accessing Properties
- 12.5 Dynamic Properties

Table of Contents

12.6 MetaTypes, Declaring, and Registering

12.7 invokeMethod()

12.8 Exercises: Reflection

12.9 Review Questions

Chapter 13: Models and Views

13.1 Model-View-Controller (MVC)

13.2 Qt Models and Views

13.3 Table Models

13.4 Tree Models

13.5 Smarter Pointers

13.6 Exercises: Models and Views

13.7 Review Questions

Chapter 14: Validation and Regular Expressions

14.1 Input Masks

14.2 Validators

14.3 Regular Expressions

14.4 Regular Expression Validation

14.5 Subclassing QValidator

14.6 Exercises: Validation and Regular Expressions

14.7 Review Questions

Chapter 15: Parsing XML

15.1 The Qt XML Parsers

15.2 SAX Parsing

15.3 XML, Tree Structures, and DOM

15.4 XML Streams

15.5 Review Questions

Chapter 16: More Design Patterns

16.1 Creational Patterns

16.2 Memento Pattern

Table of Contents

16.3 Façade Pattern

16.4 Review Questions

Chapter 17: Concurrency

17.1 QProcess and Process Control

17.2 QThread and QtConcurrent

17.3 Exercises: QThread and QtConcurrent

17.4 Review Questions

Chapter 18: Database Programming

18.1 QSqlDatabase: Connecting to SQL from Qt

18.2 Queries and Result Sets

18.3 Database Models

18.4 Review Questions

Part II: C++ Language Reference

Chapter 19: Types and Expressions

19.1 Operators

19.2 Statements and Control Structures

19.3 Evaluation of Logical Expressions

19.4 Enumerations

19.5 Signed and Unsigned Integral Types

19.6 Standard Expression Conversions

19.7 Explicit Conversions

19.8 Safer Typecasting Using ANSI C++ Typecasts

19.9 Overloading Special Operators

19.10 Runtime Type Identification

19.11 Member Selection Operators

19.12 Exercises: Types and Expressions

19.13 Review Questions

Chapter 20: Scope and Storage Class

Table of Contents

20.1 Declarations and Definitions

20.2 Identifier Scope

20.3 Storage Class

20.4 Namespaces

20.5 Review Questions

Chapter 21: Memory Access

21.1 Pointer Pathology

21.2 Further Pointer Pathology with Heap Memory

21.3 Memory Access Summary

21.4 Introduction to Arrays

21.5 Pointer Arithmetic

21.6 Arrays, Functions, and Return Values

21.7 Different Kinds of Arrays

21.8 Valid Pointer Operations

21.9 Arrays and Memory: Important Points

21.10 Exercises: Memory Access

21.11 Review Questions

Chapter 22: Inheritance in Detail

22.1 virtual Pointers, virtual Tables

22.2 Polymorphism and virtual Destructors

22.3 Multiple Inheritance

22.4 public, protected, and private Derivation

22.5 Review Questions

Part III: Programming Assignments

Chapter 23: MP3 Jukebox Assignments

23.1 Phonon/MultiMediaKit Setup

23.2 Playlist

23.3 Playlists

23.4 Source Selector

Table of Contents

23.5 Database Playlists

23.6 Star Delegates

23.7 Sorting, Filtering, and Editing Playlists

Appendix A: C++ Reserved Keywords

Appendix B: Standard Headers

Appendix C: Development Tools

Appendix D: Alans Quick Start Guide to Debian for
Programmers

Appendix E: C++/Qt Setup

Bibliography

Index