

全新打造的Qt爱好者社区强力支持！  
短期即可入门及提高的优秀Qt系列书籍！

Qt应用编程系列丛书



霍亚飞 编著  
吴迪 白建平 董世明 审校

# Qt及Qt Quick 开发实战精解

- **全新**，基于最新的Qt及Qt Creator编写，包含Qt Quick！
- **经典**，基于经典的Qt网络博客编写，可无限更新！
- **综合**，对众多知识点进行综合应用，实例经典实用！
- **系统**，与《Qt Creator快速入门》配套，理论结合实际！



北京航空航天大学出版社  
BEIHANG UNIVERSITY PRESS

## 内 容 简 介

本书主要讲解了5个Qt综合应用程序的开发过程和Qt Quick的相关内容。本书内容主要包括两部分:第一部分是多文档编辑器、方块游戏、音乐播放器、数据管理系统、局域网聊天工具这5个实用的Qt实例的详细讲解;第二部分是Qt Quick技术的全面介绍。

本书的内容全面、实用,讲解通俗易懂,适合有一定Qt基础并且想学习Qt综合实例开发或者想学习Qt Quick技术的读者。对于没有Qt基础的读者,可以先学习《Qt Creator 快速入门》一书。

### 图书在版编目(CIP)数据

Qt 及 Qt Quick 开发实战精解 / 霍亚飞编著. — 北京:  
北京航空航天大学出版社, 2012. 5  
ISBN 978 - 7 - 5124 - 0781 - 7

I. ①Q… II. ①霍… III. ①移动终端—应用程序—  
程序设计 IV. ①TN929.53

中国版本图书馆 CIP 数据核字(2012)第 063827 号

版权所有,侵权必究。

### Qt 及 Qt Quick 开发实战精解

霍亚飞 编著

吴迪 白建平 董世明 审校

责任编辑 董立娟

\*

北京航空航天大学出版社出版发行

北京市海淀区学院路37号(邮编100191) <http://www.buaapress.com.cn>

发行部电话:(010)82317024 传真:(010)82328026

读者信箱:emsbook@gmail.com 邮购电话:(010)82316936

涿州市新华印刷有限公司印装 各地书店经销

\*

开本:710×1 000 1/16 印张:17 字数:372千字

2012年5月第1版 2012年5月第1次印刷 印数:4 000册

ISBN 978 - 7 - 5124 - 0781 - 7 定价:36.00元

若本书有倒页、脱页、缺页等印装质量问题,请与本社发行部联系调换。联系电话:010 - 82317024

# 丛书前言

---

本系列书籍的原型是网络上的一系列博客教程。2009年,当作者开始使用推出不久的 Qt 开发环境 Qt Creator 时,由于该软件还不成熟,所以经常出现一些问题,但是在网络上却几乎找不到任何的相关内容可以参考。于是作者便将自己的学习经验和心得进行整理,以 yafeilinux 为网名在博客中推出了一系列 Qt 和 Qt Creator 相关教程。这些教程浅显易懂,使用的描述语言很自然,而且图文并茂,每一个知识点都设计了一个可以运行的示例程序。由于这些特点都很符合读者的需求,所以这个系列教程得到了众多网友的肯定和称赞,这也给了作者很大的信心继续写下去。到 2010 年,作者接到北航出版社的邀请,希望可以将系列教程整理成书。由于当时想写成一本与众不同的书籍,要与市面上已经出版的书籍产生差异,所以一直等待着 Qt 和 Qt Creator 全新版本的发布,直到现在才完成本套书籍的编写。

在这一年里,作者创建了 yafeilinux.com 网站,希望可以让更多的人一起参与到 Qt 的普及工作中。在该网站创建后,作者不仅将系列教程扩展到了 Qt 基础、2D 绘图、数据库操作和网络通信等几个部分,而且还推出了多个专题教程和开源软件。其中,《Qt 串口通信专题教程》在网络上广为流传,几乎成为学习 Qt 串口编程的必备教程;而开源软件中的“音乐播放器”,在推出不到一个月的时间里就有了上万次的下载量。现在网站的访问量已经超过百万,同时在作者的邮箱中,一年里收到了两千多封请教和交流的邮件,而 QQ 交流群更是建一个满一个。这些都让作者看到了 Qt 爱好者的热情和 Qt 发展的良好趋势,也让作者下定决心要写出一本好书。2011 年 3 月,Qt 4.7.2 和 Qt Creator 2.1.0 正式发布,该版本中 Qt Creator 第一次正式使用了中文界面,也是第一次正式支持 Qt Quick 编程。作者感觉是时候开始正式写作了,有了这一年来和众多网友的交流,以及作者在 Qt 各个应用方面的编程积累,已经有了足够的信心来写出一部经典作品。

直至今日书已成型,10 个月的写作包含了太多的心酸与喜悦,最终呈现给读者的是包括了《Qt Creator 快速入门》和《Qt 及 Qt Quick 开发实战精解》两部作品的一个系书籍。之所以同时推出两部作品,而没有将其合成一部书籍,是为了给不同学习阶段的读者提供灵活的选择。尽管本个系书籍已经包含了 Qt 中最常用的大部分内容,不过 Qt 实在太庞大了,所以还是有很多内容没有写进去,这些会在 yafeilinux.com 网站中得到补充。一本书的厚度是有限的,但是网络中的内容却可以是无限的,

网上的系列教程,作者还会一直写下去的。

## Qt 简介

Qt 是一个跨平台应用程序和 UI 开发框架。使用 Qt 只需一次性开发应用程序,无须重新编写源代码,便可跨不同桌面和嵌入式操作系统部署这些应用程序。Qt Software 的前身为创始于 1994 年的 Trolltech(奇趣科技),Trolltech 于 2008 年 6 月被 Nokia 收购,加速了其跨平台开发战略。Qt Creator 是全新的跨平台 Qt IDE,可单独使用,也可与 Qt 库和开发工具组成一套完整的 SDK。其中包括:高级 C++ 代码编辑器、项目和生成管理工具、集成的上下文相关的帮助系统、图形化调试器、代码管理和浏览工具。Qt Quick 是在 Qt 4.7 中被引进的一种高级用户界面技术,开发人员和设计人员可用它协同创建动画触摸式用户界面和应用程序。

## 系列丛书特色

本系列丛书作为全面介绍 Qt、Qt Creator 和 Qt Quick 的入门级教材,也是市面上第一套详细介绍 Qt Creator 和 Qt Quick 的教材。与其他相关书籍最大的不同之处还在于,本套书籍是基于网络博客教程的。综合来说,本套书籍主要具有以下特色:

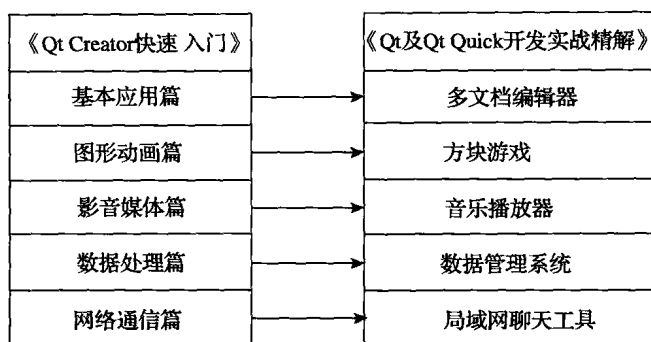
- ▶ 最新。基于最新的 Qt 4.7.2 和 Qt Creator 2.1.0 版本进行编写,在该版本中首次正式集成了 Qt Quick。
- ▶ 最全。书中的内容包含了 Qt 基础、图形动画、多媒体、数据库、网络通信、WebKit 以及 Qt Quick 等所有基本的应用内容。
- ▶ 无限更新。本套书籍对应的网络教程是无限更新的,而且读者可以通过邮件、QQ 群或者论坛和作者零距离接触。
- ▶ 按应用分类。本套书籍中将同一应用的相关章节放在一起作为一个篇章,例如图形动画篇中讲解了所有与绘图和动画相关的知识,这样组织更有利于读者在程序开发时进行选择参考学习。
- ▶ 知识点讲解与大型实例设计分离。本套书籍将知识的讲解和综合实例的设计分为了两部分。这样读者便不会因为要在很长的代码中学习某个知识点而一头雾水,但每一个知识点都设计了一个很简单的示例程序,让读者可以实际地看到该知识点的应用效果;而设计大型实例时,将主要的精力放在程序框架和功能实现上,不再细讲其中的知识点,这样更有利于读者学习设计综合性程序。
- ▶ 程序设计分步实现。本套书籍中不是像其他书籍那样将一个完整示例程序的所有代码全部列出并进行讲解,而是分步来设计该程序,从创建项目开始,一

个功能模块一个功能模块的分步添加,这样可以使读者更好地了解程序的设计过程。

- ▶ 关键字提示。本套书籍的编写主要基于 Qt 参考文档,而且所讲解的知识点也只是 Qt 参考文档中的部分内容,读者在学习时一定要多参考 Qt 帮助文档。在本套书籍讲解的所有知识点和示例程序中,都很明显地标出了其在 Qt 帮助中对应的关键字,这样可以使读者对书中的内容有迹可循。

## 丛书结构

本套书籍一共分为两本:《Qt Creator 快速入门》和《Qt 及 Qt Quick 开发实战精解》。《Qt Creator 快速入门》中主要进行知识点的详细讲解,根据应用的不同分为了 5 个篇章:基本应用、图形动画、影音媒体、数据处理和网络通信;在《Qt 及 Qt Quick 开发实战精解》中主要讲解了 5 个综合实例以及 Qt Quick 的内容,其中的 5 个实例程序分别对应了《Qt Creator 快速入门》的 5 个篇章,也就是说每一个应用实例都是对应篇章知识点的综合应用。本套书籍的结构如下图所示:



## 使用本系列丛书籍

对于没有任何 Qt 编程经验的读者,建议先学习《Qt Creator 快速入门》。该书对 Qt 基本内容以及 Qt Creator 开发环境进行了详细讲解,而且每一个知识点都设计了一个可以独立运行的示例程序,非常适合初学者入门使用。对于有一定 Qt 编程经验,并且想学习 Qt 综合实例编写的读者,推荐使用《Qt 及 Qt Quick 开发实战精解》。在该书中对于不同的应用领域分别设计了一个实用的应用程序,而且每一个实例都应用了众多的知识点,非常适合没有实际开发经验的读者。如果读者想学习全新的 Qt Quick 技术,那么《Qt 及 Qt Quick 开发实战精解》也是很好的选择,该书的 Qt Quick 部分详细介绍了 Qt Quick 的方方面面。

当然,本套书籍是一个相互联系的整体,建议读者最好可以在学习《Qt Creator

快速入门》的同时学习《Qt 及 Qt Quick 开发实战精解》，每学完一篇内容就去学习对应的实例程序，这样可以达到更好的效果。

在学习过程中要多动手，尽量按照步骤编写代码，只有在遇到自己无法解决的问题时再去参考本系列书籍提供的源代码。每当学习一个知识点时，书中都会给出 Qt 帮助中的关键字，建议参考 Qt 的帮助文档，看英文原文是怎么描述的。不要害怕去看那些英文文档，因为不可能在网上找到所有文档的中文翻译，而且有些翻译也偏离了原意，所以最终还是要自己去看英文的参考文档的。学会看参考文档是入门 Qt 编程的重要一步，不要怕自己英文不好，其实跟自己的英文水平关系不大，只要坚持，掌握了一些英文术语和关键词以后，阅读英文文档是不成问题的。

## Qt 版本说明

本系列书籍是基于 Windows 下的 Qt 4.7.2 和 Qt Creator 2.1.0 版本的，这两个版本是在本系列书籍开始编写时的最新版本。为了避免读者使用不同的操作系统而产生不必要的问题，本系列书籍采用了最常用的 Windows XP 系统。并且本系列书籍中只讲解了 Qt 桌面编程，并没有涉及移动平台开发和嵌入式开发的内容，如果要进行移动平台(Symbian 和 MeeGo Harmattan 系统)软件开发，那么可以下载 Qt SDK，它集成了 Qt 桌面、移动平台的开发库以及 Qt Creator 开发环境。

这里要向对 Qt 版本不是很了解的读者说明一下，对于 Qt 程序开发，无论是在 Windows 系统下进行开发，还是在 Linux 系统下进行开发；无论是进行桌面程序开发，还是进行移动平台或者嵌入式平台的开发，只需要编写一次代码，然后分别进行编译就可以了。这就是 Qt 最大的特点，即所谓的“一次编写，随处编译”。也就是说，读者只需要学好本系列书籍中 Qt 的基本内容，然后编写代码，使用 Qt 不同的版本进行编译即可。

在学习本系列书籍时，推荐读者使用指定的 Qt 和 Qt Creator 版本，因为对于初学者来说，任何微小的差异都可能导致错误的理解。当然，也可以使用其他版本，在作者的网站上有不同平台的各个版本的使用教程。

## 致 谢

感谢北京航空航天大学出版社的相关人员，是他们的邀请和支持，才让作者坚定信心要写一本与众不同的经典作品。

感谢那些关注和爱好 Qt 的朋友们，是他们的支持和帮助，才让作者一步一步走下来。其中一些朋友还参与了本系列书籍的审校和代码审核工作，具体的审校分工是：解放军装甲兵工程学院的吴迪(wd007)完成了初审、Qt 中文论坛管理员白建平(XChinux)完成了二审、上海大学计算机工程与科学学院的董世明完成了终审；参加

代码审核的朋友有：程梁(豆子)、刘柏桑(紫侠)、么贺文(mehewen)、胡峰(古月行云)、周慧宗、杨凡(云帆)和黄金金等，他们分别完成了本套书籍代码在 Windwos XP 系统、Windows 7 系统和 Ubuntu Linux 系统下的代码审核工作。是众多朋友的认真工作，才使得本系列丛书籍可以较早地出版。

由于作者技术水平有限，经验也很欠缺，再加上 Qt 的内容繁多，并且没有统一的中文术语参考，所以书中难免有各种错误理解和代码设计缺陷，恳请读者批评指正。读者可以到作者的网站 [www.yafeilinux.com](http://www.yafeilinux.com) 下载本套书籍的源码，查看与本套书籍对应的不断更新的系列教程。也可以到 Qt 爱好者社区([www.qter.org](http://www.qter.org))在本系列丛书的专版进行讨论交流。

霍亚飞  
2012 年 2 月于北京

# 前言

本书主要讲解了 5 个 Qt 综合应用程序的开发过程和 Qt Quick 的相关内容,适合有一定 Qt 基础并且想学习 Qt 综合实例开发或者想学习 Qt Quick 技术的读者。本书中的实例部分没有涉及太多知识点的讲解,而是将重点放在了知识的应用上。与本书对应的《Qt Creator 快速入门》,该书主要讲解 Qt 的知识点,本书中的几个实例就是对该书讲解的知识的综合应用。对于没有 Qt 基础,或者在学习本书实例遇到困难时,建议先学习《Qt Creator 快速入门》中相应的基础知识。

本书内容共包含了 6 章,分为两个部分:综合实例和 Qt Quick。综合实例部分共包含 5 章,分别对应《Qt Creator 快速入门》的每一个篇章设计了一个综合实例程序:

- ▶ 第 1 章 多文档编辑器:对应基本应用篇。该实例实现了一个可以同时处理多个子文档的编辑器软件,主要包含了菜单设计、主窗口应用和各功能模块间相互约束等内容。
- ▶ 第 2 章 方块游戏:对应图形动画篇。该实例实现了一个经典的俄罗斯方块游戏,主要包含了游戏逻辑设计、图形视图应用、动画设置和添加背景音乐等内容。
- ▶ 第 3 章 音乐播放器:对应影音媒体篇。该实例实现了一个可以显示桌面歌词的音乐播放器软件,主要包含了 Phonon 框架应用、文件解析、2D 绘图和系统托盘图标等内容。
- ▶ 第 4 章 数据管理系统:对应数据处理篇。该实例实现了一个使用数据库管理数据并使用饼状图显示数据的管理系统,主要包含了数据库、XML 和自定义视图等内容。
- ▶ 第 5 章 局域网聊天工具:对应网络通信篇。该实例实现了一个可以在局域网中发送信息和文件的通信工具软件,主要包含了 UDP 编程、TCP 编程和富文本处理等内容。

Qt Quick 部分只包含了第 6 章,主要讲解了 Qt 最新引入的 Qt Quick 技术,虽然 Qt Quick 编程与《Qt Creator 快速入门》中讲述的 Qt C++ 编程有着千丝万缕的联系,但是它是以一种全新的技术推出的,其中还包含了一门全新的语言 QML。



# 目 录

## 第一部分 综合实例

<b>第 1 章 多文档编辑器</b> .....	3
1.1 界面设计 .....	4
1.2 创建子窗口类 .....	5
1.3 实现菜单的功能.....	12
1.4 完善程序功能.....	21
1.5 小 结.....	26
<b>第 2 章 方块游戏</b> .....	27
2.1 方块游戏架构.....	27
2.2 实现游戏逻辑.....	29
2.3 游戏优化.....	42
2.4 小 结.....	51
<b>第 3 章 音乐播放器</b> .....	52
3.1 播放器整体架构.....	52
3.2 实现音乐播放.....	53
3.3 实现播放列表.....	59
3.4 实现桌面歌词.....	68
3.5 添加系统托盘图标.....	77
3.6 小 结.....	78
<b>第 4 章 数据管理系统</b> .....	79
4.1 功能介绍与界面设计.....	80
4.2 实现商品管理功能.....	82
4.3 显示销售统计图表.....	93
4.4 添加登录界面.....	98
4.5 小 结 .....	100
<b>第 5 章 局域网聊天工具</b> .....	101
5.1 界面设计 .....	102
5.2 实现聊天功能 .....	104

5.3 实现文件传输功能 .....	111
5.4 完善程序功能 .....	122
5.5 小 结 .....	127

## 第二部分 Qt Quick

<b>第 6 章 Qt Quick</b> .....	131
6.1 初识 QML .....	131
6.2 QML 概念及框架 .....	138
6.3 QML 中的布局管理 .....	175
6.4 基本可视元素 .....	181
6.5 事件处理 .....	191
6.6 图像、状态和动画 .....	202
6.7 QML 中的模型/视图 .....	224
6.8 QML 和 C++ 混合编程 .....	239
6.9 使用 Qt Quick 设计器 .....	251
6.10 小 结 .....	258
<b>参考文献</b> .....	259

---

# 第一部分

## 综合实例

---



# 第 1 章 多文档编辑器

这一章的例子是对《Qt Creator 快速入门》基础应用篇各章节知识的综合应用，也是一个规范的实例程序。之所以说其规范，是因为在这个程序中，我们对菜单什么时候可用/什么时候不可用、关闭程序时应该先保存已修改且尚未保存的文件等细节都做了严格的约束。而一个真正实用的应用程序，也就应该如此。

本章应用了基础篇的众多知识点，但这里只是讲解程序流程与框架，不会涉及太多知识细节的讲解，所以希望读者可以学完基础应用篇后再来学习本章。这个实例主要是对主窗口部件的应用，所以可以学完《Qt Creator 快速入门》的前 5 章再来学习本章，这样可以达到更好的效果。该实例是基于 Qt 中的 MDI Example 示例程序的，它在 Main Windows 分类下。这个程序就是以 QMainWindow 类为主窗口，以 QMdiArea 类为多文档区域，以 QTextEdit 类为子窗口部件，从而实现了一个多文档编辑器的应用。最终的运行效果如图 1-1 所示。

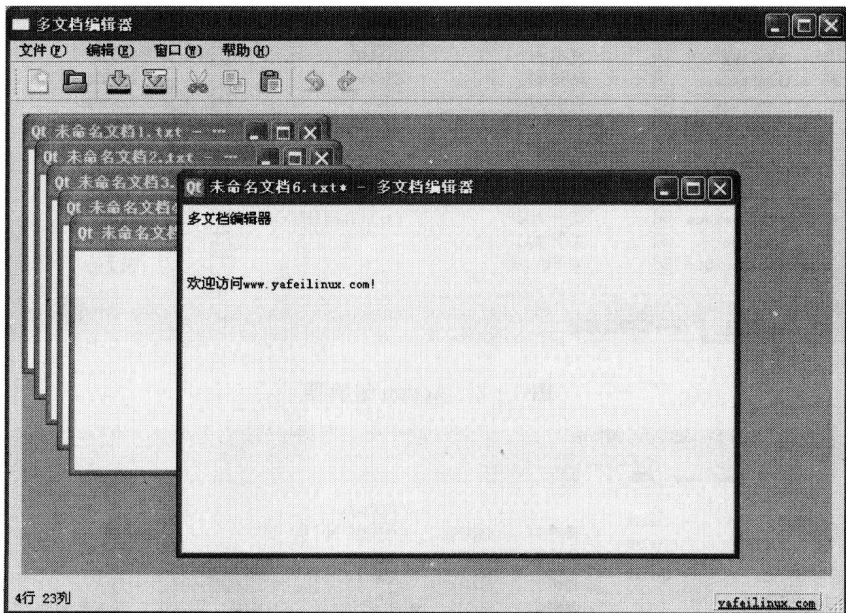


图 1-1 多文档编辑器界面

## 1.1 界面设计

首先进行界面的设计,这里主要是对主窗口菜单栏和工具栏的设计。打开 Qt Creator,创建新的项目。(项目源码路径:src\1\1-1\myMdi)新建 Qt Gui 应用,项目名称 myMdi,类名默认为 MainWindow,基类默认为 QMainWindow 都不做改动。完成后双击 mainwindow.ui 文件进入设计模式,然后添加各个菜单,所有的菜单动作如图 1-2 所示,最终的菜单栏和工具栏如图 1-3 所示。设计菜单时,如果将来触发这个菜单会弹出一个对话框进行详细设置,那么就在这个菜单文本后面添加“...”号,例如这里的“打开文件”菜单和“另存为”菜单。这里还要注意,添加动作时,一定要使动作名称和这里的 Action 编辑器中所使用的名称保持一致,因为在后面的程序中还要用到它们。

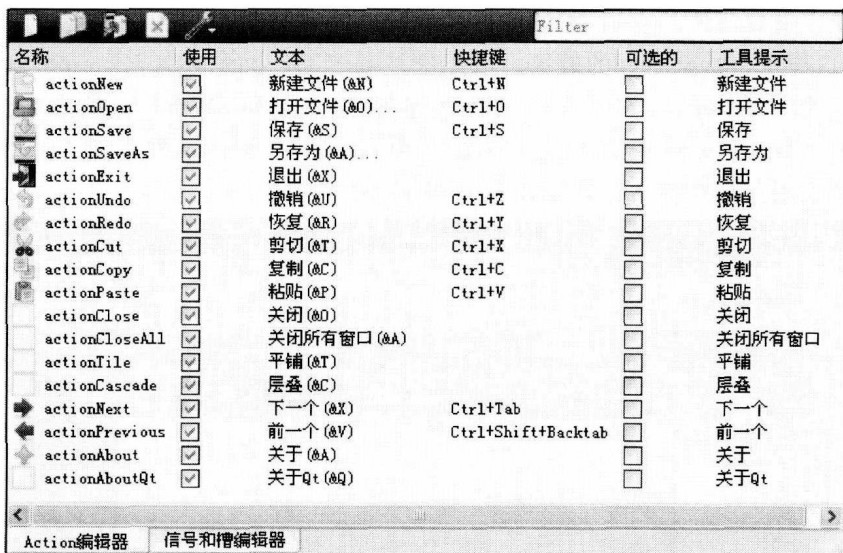


图 1-2 Action 编辑器

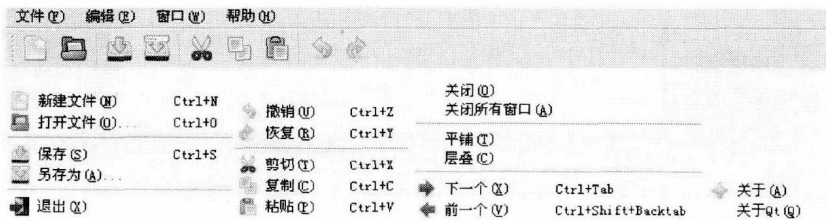


图 1-3 菜单栏与工具栏

设计完菜单栏与工具栏后,向主窗口中心区域拖入一个 MdiArea 部件,并单击主窗口界面,按下 Ctrl+G 快捷键,使其处于栅格布局之中。可以看一下对象列表窗

口,确保 MdiArea 部件的 objectName 是 mdiArea,而文件菜单、编辑菜单、窗口菜单和帮助菜单的 objectName 分别是 menuF、menuE、menuW 和 menuH;如果不是,需要在属性栏中更改,因为后面的程序中要用到。

## 1.2 创建子窗口类

为了实现多文档操作,需要向 QMdiArea 中添加子窗口,而为了更好地操作子窗口,必须子类化子窗口的中心部件。因为这里子窗口的中心部件使用了 QTextEdit 类,所以要实现自己的类,它必须继承自 QTextEdit,然后在其中添加我们的功能函数。

(项目源码路径:src\1\1-2\myMdi)往项目中添加新文件,模板选择“C++类”,类名为 MdiChild,基类为 QTextEdit,类型信息选择“继承自 QWidget”。完成后在 mdichild.h 文件中添加代码:

```
class MdiChild : public QTextEdit
{
    Q_OBJECT
public:
    explicit MdiChild(QWidget *parent = 0);
    void newFile(); // 新建操作
    bool loadFile(const QString &fileName); // 加载文件
    bool save(); // 保存操作
    bool saveAs(); // 另存为操作
    bool saveFile(const QString &fileName); // 保存文件
    QString userFriendlyCurrentFile(); // 提取文件名
    QString currentFile(){return curFile;} // 返回当前文件路径

protected:
    void closeEvent(QCloseEvent * event); // 关闭事件

private slots:
    void documentWasModified(); // 文档被更改时,窗口显示更改状态标志

private:
    bool maybeSave(); // 是否需要保存
    void setCurrentFile(const QString &fileName); // 设置当前文件
    QString curFile; // 保存当前文件路径
    bool isUntitled; // 作为当前文件是否被保存到硬盘上的标志
};
```

这里在头文件中声明了 11 个函数,定义了两个变量。其中,currentFile()函数返回当前的文件路径,只有一行代码,就直接在这里定义了。所以真正需要设计的只

有 10 个函数,还有 `curFile` 与 `isUntitled` 两个变量,分别用于保存当前文件的路径和作为文件是否被保存过的标志。因为对于所有的应用程序,只有涉及新建、保存和关闭等操作时,都是使用的这些函数进行设置的,它们是一个整体,所以这里要将它们同时罗列出来。这些函数主要完成了下面几个操作:

### 1. 新建文件操作 `newFile()`

- ▶ 设置窗口编号;
- ▶ 设置文件未被保存过“`isUntitled = true;`”;
- ▶ 保存文件路径,给 `curFile` 赋初值;
- ▶ 设置子窗口标题;
- ▶ 关联文档内容改变信号到显示文档更改状态标志槽 `documentWasModified()`。

### 2. 加载文件操作 `loadFile()`

- ▶ 打开指定的文件,并读取文件内容到编辑器;
- ▶ 设置当前文件 `setCurrentFile()`,该函数可以获取文件路径,完成文件和窗口状态的设置;
- ▶ 关联文档内容改变信号到显示文档更改状态标志槽 `documentWasModified()`。

### 3. 保存操作 `save()`

- ▶ 如果文件没有被保存过(用 `isUntitled` 判断),执行另存为操作 `saveAs()`;
- ▶ 否则直接保存文件 `saveFile()`,该函数先打开指定文件,然后将编辑器的内容写入该文件,最后设置当前文件 `setCurrentFile()`。

### 4. 另存为操作 `saveAs()`

- ▶ 从文件对话框获取文件路径;
- ▶ 如果路径不为空,则保存文件 `saveFile()`。

### 5. 关闭操作 `closeEvent()`

- ▶ 如果 `maybeSave()` 函数返回为真,则关闭窗口。`maybeSave()` 函数判断文档是否被更改过,如果被更改过,则弹出对话框,让用户选择是否保存更改,或者取消关闭操作。如果用户选择保存更改,则返回保存操作 `save()` 的结果,如果选择取消,则返回 `false`。否则,直接返回 `true`。
- ▶ 如果 `maybeSave()` 函数返回为假,则忽略该事件。

下面到 `mdichild.cpp` 文件中添加代码。这里先列出所有需要添加的头文件,当然,实际写程序时,是用到哪个类才去添加那个类的头文件的。

```
#include <QFile>
#include <QTextStream>
#include <QMessageBox>
#include <QFileInfo>
#include <QApplication>
```



```
# include <QFileDialog>
# include <QCloseEvent>
# include <QPushButton>
```

然后在 MdiChild 类的构造函数中添加两行代码：

```
// 设置在子窗口关闭时销毁这个类的对象
setAttribute(Qt::WA_DeleteOnClose);

// 初始 isUntitled 为 true
isUntitled = true;
```

第一行代码会在后面讲解关闭操作时提到，而第二行代码只是对变量的初始化。下面开始介绍 5 个操作的函数实现。

首先是新建文件操作：

```
void MdiChild::newFile()
{
    // 设置窗口编号,因为编号一直被保存,所以需要使用静态变量
    static int sequenceNumber = 1;

    // 新建的文档没有被保存过
    isUntitled = true;

    // 将当前文件命名为未命名文档加编号,编号先使用再加 1.
    curFile = tr("未命名文档 %1.txt").arg(sequenceNumber + +);

    // 设置窗口标题,使用[*]可以在文档被更改后在文件名称后显示"*"号
    setWindowTitle(curFile + "[*]" + tr(" - 多文档编辑器"));

    // 文档更改时发射 contentsChanged()信号,执行 documentWasModified()槽
    connect(document(), SIGNAL(contentsChanged()),
            this, SLOT(documentWasModified()));
}
```

这里在设置窗口标题时添加了“[\*]”字符,它可以保证编辑器内容被更改后,在相应位置显示“\*”号。下面是 documentWasModified()槽的定义：

```
void MdiChild::documentWasModified()
{
    // 根据文档的 isModified()函数的返回值,判断编辑器内容是否被更改了
    // 如果被更改了,就要在设置了[*]号的地方显示“*”号,这里会在窗口标题中显示
    setWindowModified(document()->isModified());
}
```

编辑器内容是否被更改,可以使用 QTextDocument 类的 isModified()函数获知,这里使用了 QTextEdit 类的 document()函数来获取它的 QTextDocument 类对