

11 界面文件与界面类

基于C++语言的面向对象编程，离不开类的声明、定义和实例化。在之前利用Qt创造器创建的Calculator2项目中，已经包含了用于描述用户界面的界面文件，以及与之相对应的界面类，并将该类实例化为表示界面实体的C++对象。

11.1 界面文件

C:\Users\Minwei\Projects\Qt\Calculator2\calculatordialog.ui:

```
1 <?xml version="1.0" encoding="UTF-8"?>
2 <ui version="4.0">
3   <class>CalculatorDialog</class>
4   <widget class="QDialog" name="CalculatorDialog">
5     <property name="geometry">
6       <rect>
7         <x>0</x>
8         <y>0</y>
9         <width>499</width>
10        <height>43</height>
11      </rect>
12    </property>
13    <property name="windowTitle">
14      <string>计算器</string>
15    </property>
16    <layout class="QHBoxLayout" name="m_layout">
17      <item>
18        <widget class="QLineEdit" name="m_editX">
19          <property name="alignment">
20            <set>Qt::AlignRight|Qt::AlignTrailing|Qt::AlignVCenter</set>
21          </property>
22        </widget>
23      </item>
24      <item>
25        <widget class="QLabel" name="m_label">
26          <property name="text">
27            <string>+</string>
28          </property>
29          <property name="alignment">
30            <set>Qt::AlignCenter</set>
31          </property>
32        </widget>
33      </item>
34      <item>
35        <widget class="QLineEdit" name="m_editY">
36          <property name="alignment">
37            <set>Qt::AlignRight|Qt::AlignTrailing|Qt::AlignVCenter</set>
38          </property>
39        </widget>
40      </item>
41      <item>
42        <widget class="QPushButton" name="m_button">
43          <property name="enabled">
44            <bool>>false</bool>
```

```

45     </property>
46     <property name="text">
47         <string>=</string>
48     </property>
49 </widget>
50 </item>
51 <item>
52     <widget class="QLineEdit" name="m_editz">
53         <property name="alignment">
54             <set>Qt::AlignRight|Qt::AlignTrailing|Qt::AlignVCenter</set>
55         </property>
56         <property name="readOnly">
57             <bool>true</bool>
58         </property>
59     </widget>
60 </item>
61 </layout>
62 </widget>
63 <resources/>
64 <connections/>
65 </ui>

```

界面文件的扩展名为“.ui”，由Qt设计师自动生成，将程序编写者在图形化界面设计器中，所做的所有设计工作，全部记录在一个XML格式的文本文件中。该文件可被Qt设计师重新打开，并编辑修改。

11.2 原始界面类

Qt创造器调用界面编译器uic，执行如下命令：

```
1 uic CalculatorDialog.ui -o ui_CalculatorDialog.h
```

将XML格式的界面文件编译为原始界面类文件。

C:\Users\Minwei\Projects\Qt\Calculator2\ui_calculatordialog.h:

```

1  /*****
2  ****
3  ** Form generated from reading UI file 'calculatordialog.ui'
4  **
5  ** Created by: Qt User Interface Compiler version 5.12.8
6  **
7  ** WARNING! All changes made in this file will be lost when recompiling UI
8  ** file!
9  ****/
10
11
12 #ifndef UI_CALCULATORDIALOG_H
13 #define UI_CALCULATORDIALOG_H
14
15 #include <QtCore/QVariant>
16 #include <QtWidgets/QApplication>
17 #include <QtWidgets/QDialog>
18 #include <QtWidgets/QHBoxLayout>
19 #include <QtWidgets/QLabel>
20 #include <QtWidgets/QLineEdit>

```

```

18 #include <QtWidgets/QPushButton>
19
20 QT_BEGIN_NAMESPACE
21
22 class Ui_CalculatorDialog
23 {
24 public:
25     QHBoxLayout *m_layout;
26     QLineEdit *m_editX;
27     QLabel *m_label;
28     QLineEdit *m_editY;
29     QPushButton *m_button;
30     QLineEdit *m_editZ;
31
32     void setupUi(QDialog *calculatorDialog)
33     {
34         if (calculatorDialog->objectName().isEmpty())
35             calculatorDialog-
36 >setObjectName(QString::fromUtf8("calculatorDialog"));
37         calculatorDialog->resize(499, 43);
38         m_layout = new QHBoxLayout(calculatorDialog);
39         m_layout->setObjectName(QString::fromUtf8("m_layout"));
40         m_editX = new QLineEdit(calculatorDialog);
41         m_editX->setObjectName(QString::fromUtf8("m_editX"));
42         m_editX-
43 >setAlignment(Qt::AlignRight|Qt::AlignTrailing|Qt::AlignVCenter);
44
45         m_layout->addWidget(m_editX);
46
47         m_label = new QLabel(calculatorDialog);
48         m_label->setObjectName(QString::fromUtf8("m_label"));
49         m_label->setAlignment(Qt::AlignCenter);
50
51         m_layout->addWidget(m_label);
52
53         m_editY = new QLineEdit(calculatorDialog);
54         m_editY->setObjectName(QString::fromUtf8("m_editY"));
55         m_editY-
56 >setAlignment(Qt::AlignRight|Qt::AlignTrailing|Qt::AlignVCenter);
57
58         m_layout->addWidget(m_editY);
59
60         m_button = new QPushButton(calculatorDialog);
61         m_button->setObjectName(QString::fromUtf8("m_button"));
62         m_button->setEnabled(false);
63
64         m_layout->addWidget(m_button);
65
66         m_editZ = new QLineEdit(calculatorDialog);
67         m_editZ->setObjectName(QString::fromUtf8("m_editZ"));
68         m_editZ-
69 >setAlignment(Qt::AlignRight|Qt::AlignTrailing|Qt::AlignVCenter);
70         m_editZ->setReadOnly(true);
71
72         m_layout->addWidget(m_editZ);

```

```

70
71     retranslateUi( CalculatorDialog );
72
73     QObject::connectSlotsByName( CalculatorDialog );
74 } // setupUi
75
76 void retranslateUi( QDialog * CalculatorDialog )
77 {
78     CalculatorDialog-
79 >setWindowTitle( QApplication::translate( "CalculatorDialog",
80 "\350\256\241\347\256\227\345\231\250", nullptr ) );
81     m_label->setText( QApplication::translate( "CalculatorDialog", "+",
82 nullptr ) );
83     m_button->setText( QApplication::translate( "CalculatorDialog", "=",
84 nullptr ) );
85 } // retranslateUi
86
87 };
88
89 namespace Ui {
90     class CalculatorDialog: public Ui_CalculatorDialog {};
91 } // namespace Ui
92
93 QT_END_NAMESPACE
94
95 #endif // UI_CALCULATORIALOG_H

```

该文件定义了一个名为“Ui_CalculatorDialog”的原始界面类，其中包含了界面文件中对用户界面的所有描述。该类的公有方法setupUi，用于创建并初始化界面中的所有元素。注意，在项目的其它代码中，并没有直接使用Ui_CalculatorDialog类，而是使用了该类在Ui名字空间中的子类CalculatorDialog。

11.3 自定义界面类

原始界面类由界面编译器uic根据界面文件自动生成，不需要也不应该手动编辑修改该文件。所有针对用户界面的定制化操作，都在自定义界面类中完成。自定义界面类的声明和实现分别位于calculatordialog.h和calculatordialog.cpp文件中。

C:\Users\Minwei\Projects\Qt\Calculator2\calculatordialog.h:

```

1  #ifndef CALCULATORIALOG_H
2  #define CALCULATORIALOG_H
3
4  #include <QDialog>
5
6  QT_BEGIN_NAMESPACE
7  namespace Ui { class CalculatorDialog; }
8  QT_END_NAMESPACE
9
10 class CalculatorDialog : public QDialog
11 {
12     Q_OBJECT
13
14 public:
15     CalculatorDialog( QWidget *parent = nullptr );
16     ~CalculatorDialog();

```

```

17
18 private:
19     Ui::CalculatorDialog *ui;
20 };
21
22 #endif // CALCULATOR_DIALOG_H

```

C:\Users\Minwei\Projects\Qt\Calculator2\calculatordialog.cpp:

```

1 #include "calculatordialog.h"
2 #include "ui_calculatordialog.h"
3
4 CalculatorDialog::CalculatorDialog(QWidget *parent)
5     : QDialog(parent)
6     , ui(new Ui::CalculatorDialog)
7 {
8     ui->setupUi(this);
9 }
10
11 CalculatorDialog::~CalculatorDialog()
12 {
13     delete ui;
14 }

```

这里定义了一个名为“CalculatorDialog”的类，注意它与前面提到的，Ui名字空间里的CalculatorDialog类，不是同一个类。这里的CalculatorDialog类是QDialog类的子类，而Ui名字空间里的CalculatorDialog类则是Ui_CalculatorDialog类的子类。

CalculatorDialog类可以多重继承自QDialog类和Ui::CalculatorDialog类，比如：

```

1 class CalculatorDialog: public QDialog, public Ui::CalculatorDialog
2 {
3     ...
4     CalculatorDialog(QWidget* parent): QDialog(parent)
5     {
6         setupUi(this); // 直接调用
7     }
8     ...
9 };

```

也可以只继承QDialog类，并组合Ui::CalculatorDialog类的实例，比如：

```

1 class CalculatorDialog: public QDialog
2 {
3     ...
4     Ui::CalculatorDialog* ui;
5     ...
6     CalculatorDialog(QWidget* parent): QDialog(parent), ui(new
7     Ui::CalculatorDialog)
8     {
9         ui->setupUi(this); // 间接调用
10 }
};

```

显然，Qt创造器自动生成的代码，默认采用的是组合方式。Qt创造器所生成的自定义界面类，只包含最基本的代码框架，程序编写者将进一步完善这个类，并将业务逻辑纳入其中。

11.4 界面对象

Qt创造器已经自动生成了可执行程序的入口函数main，位于名为“main.cpp”的文件中。

C:\Users\Minwei\Projects\Qt\Calculator2\main.cpp:

```
1  #include "calculatordialog.h"
2
3  #include <QApplication>
4
5  int main(int argc, char *argv[])
6  {
7      QApplication a(argc, argv);
8      CalculatorDialog w;
9      w.show();
10     return a.exec();
11 }
```

这里直接将前面定义的界面类CalculatorDialog，实例化为一个界面对象w，并通过其show方法，显示该对象。