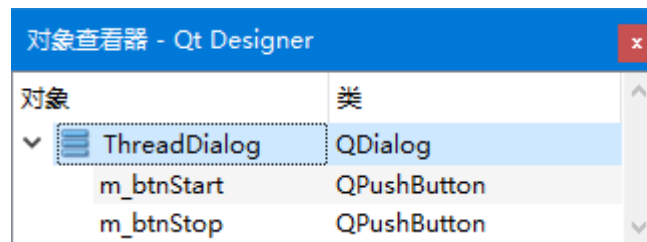
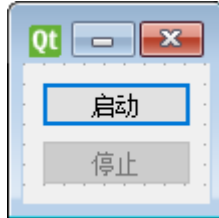


78 基于QThread的多线程编程

78.1 创建项目

通过QtCreator, 在C:\Users\Minwei\Projects\Qt路径下, 创建名为Thread的项目。

78.2 设计界面



C:\Users\Minwei\Projects\Qt\Thread\threaddialog.ui:

```
1 <?xml version="1.0" encoding="UTF-8"?>
2 <ui version="4.0">
3 <class>ThreadDialog</class>
4 <widget class="QDialog" name="ThreadDialog">
5 <property name="geometry">
6 <rect>
7 <x>0</x>
8 <y>0</y>
9 <width>93</width>
10 <height>70</height>
11 </rect>
12 </property>
13 <property name="windowTitle">
14 <string>线程</string>
15 </property>
16 <layout class="QVBoxLayout" name="m_layout">
17 <item>
18 <widget class="QPushButton" name="m_btnStart">
19 <property name="sizePolicy">
20 <sizepolicy hstretch="Preferred" vstretch="Preferred">
21 <horstretch>0</horstretch>
22 <verstretch>0</verstretch>
23 </sizepolicy>
24 </property>
25 <property name="text">
26 <string>启动</string>
27 </property>
28 <property name="default">
29 <bool>true</bool>
30 </property>
31 </widget>
```

```

32     </item>
33     <item>
34         <widget class="QPushButton" name="m_btnStop">
35             <property name="enabled">
36                 <bool>>false</bool>
37             </property>
38             <property name="sizePolicy">
39                 <sizepolicy hsize="Preferred" vsize="Preferred">
40                     <horstretch>0</horstretch>
41                     <verstretch>0</verstretch>
42                 </sizepolicy>
43             </property>
44             <property name="text">
45                 <string>停止</string>
46             </property>
47         </widget>
48     </item>
49 </layout>
50 </widget>
51 <resources/>
52 <connections/>
53 </ui>

```

78.3 实现功能

78.3.1 工作线程

C:\Users\Minwei\Projects\Qt\Thread\workthread.h:

```

1  #ifndef WORKTHREAD_H
2  #define WORKTHREAD_H
3
4  #include <QThread>
5
6  class workThread : public QThread
7  {
8  public:
9      workThread(unsigned long msec, char ch);
10
11     void stop();
12
13 protected:
14     void run();
15
16 private:
17     unsigned long m_msec;
18     char m_ch;
19     bool m_stop;
20 };
21
22 #endif // WORKTHREAD_H

```

C:\Users\Minwei\Projects\Qt\Thread\workthread.cpp:

```


```

```

1  #include <iostream>
2  using namespace std;
3
4  #include "workthread.h"
5
6  workThread::workThread(unsigned long msec, char ch)
7      : m_msec(msec)
8      , m_ch(ch)
9      , m_stop(false)
10 {
11 }
12
13 void workThread::stop()
14 {
15     m_stop = true;
16 }
17
18 void workThread::run()
19 {
20     while (!m_stop)
21     {
22         cout << m_ch << flush;
23         msleep(m_msec);
24     }
25 }

```

78.3.2 主窗口

C:\Users\Minwei\Projects\Qt\Thread\threaddialog.h:

```

1  #ifndef THREADDIALOG_H
2  #define THREADDIALOG_H
3
4  #include <QDialog>
5
6  #include "workthread.h"
7
8  QT_BEGIN_NAMESPACE
9  namespace Ui { class ThreadDialog; }
10 QT_END_NAMESPACE
11
12 class ThreadDialog : public QDialog
13 {
14     Q_OBJECT
15
16 public:
17     ThreadDialog(QWidget *parent = nullptr);
18     ~ThreadDialog();
19
20 private slots:
21     void on_m_btnStart_clicked();
22     void on_m_btnStop_clicked();
23
24 private:
25     Ui::ThreadDialog *ui;

```

```

26     workThread* m_work1;
27     workThread* m_work2;
28 };
29
30 #endif // THREADDIALOG_H

```

C:\Users\Minwei\Projects\Qt\Thread\threaddialog.cpp:

```

1  #include "threaddialog.h"
2  #include "ui_threaddialog.h"
3
4  ThreadDialog::ThreadDialog(QWidget *parent)
5      : QDialog(parent)
6        , ui(new Ui::ThreadDialog)
7        , m_work1(Q_NULLPTR)
8        , m_work2(Q_NULLPTR)
9  {
10     ui->setupUi(this);
11 }
12
13 ThreadDialog::~ThreadDialog()
14 {
15     delete ui;
16 }
17
18 void ThreadDialog::on_m_btnStart_clicked()
19 {
20     ui->m_btnStart->setEnabled(false);
21     ui->m_btnStop->setEnabled(true);
22
23     m_work1 = new workThread(500, '|');
24     m_work1->start();
25
26     m_work2 = new workThread(100, '.');
27     m_work2->start();
28 }
29
30 void ThreadDialog::on_m_btnStop_clicked()
31 {
32     m_work1->stop();
33     m_work1->wait();
34     delete m_work1;
35     m_work1 = Q_NULLPTR;
36
37     m_work2->stop();
38     m_work2->wait();
39     delete m_work2;
40     m_work2 = Q_NULLPTR;
41
42     ui->m_btnStart->setEnabled(true);
43     ui->m_btnStop->setEnabled(false);
44 }

```

