

28 反弹小球

28.1 需求分析

设计并实现一个显示反弹小球的对话框。小球在窗口内沿直线运动，碰到窗口边框即反弹，不得超出窗口范围。

28.2 创建项目

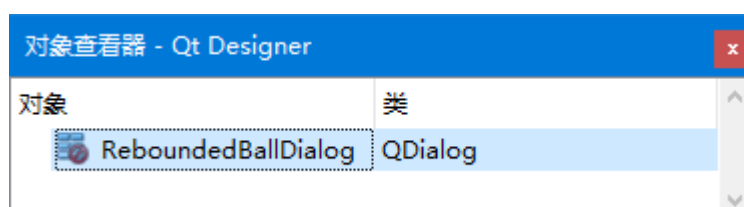
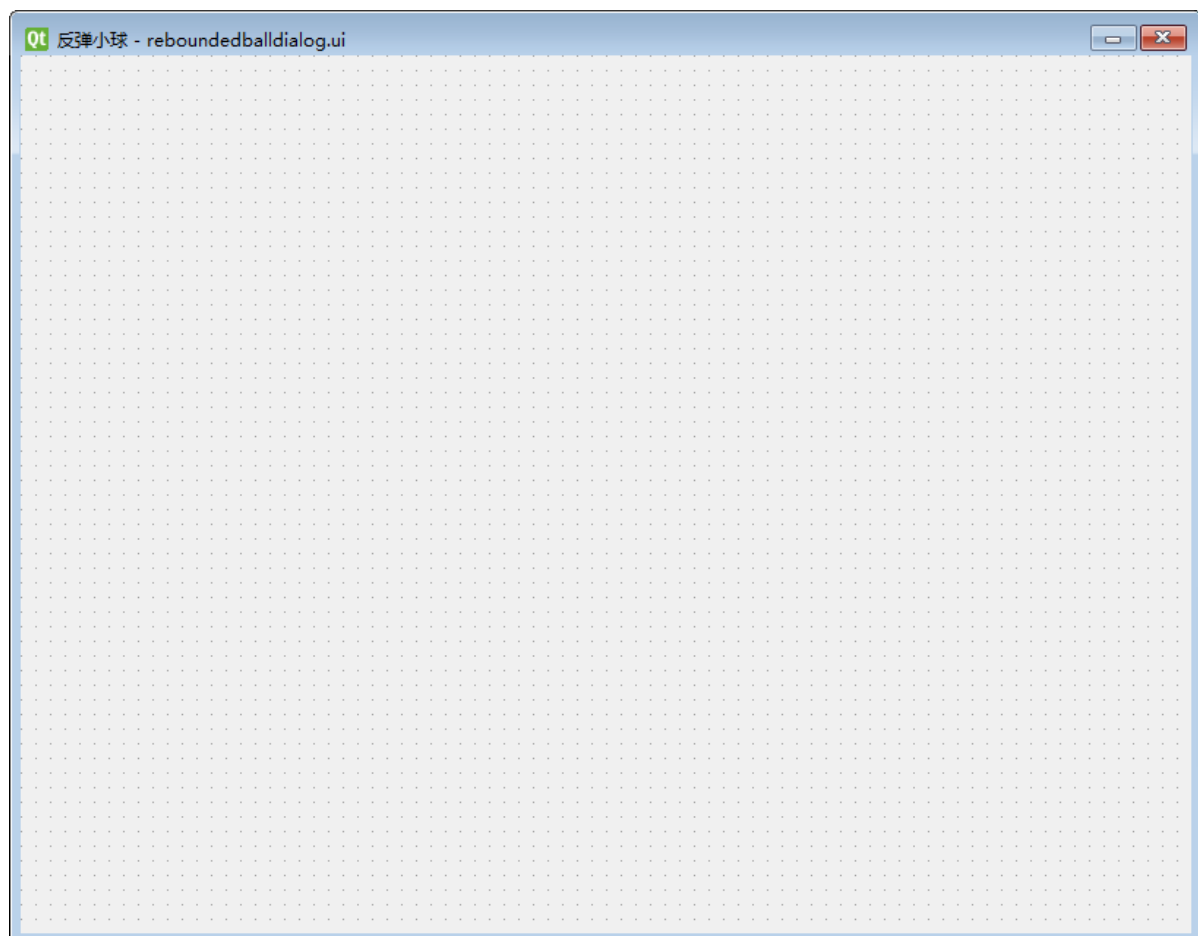
通过QtCreator，在C:\Users\Minwei\Projects\Qt路径下，创建名为ReboundedBall的项目。

28.3 添加资源

C:\Users\Minwei\Projects\Qt\ReboundedBall\ReboundedBall.qrc:

```
1 <RCC>
2     <qresource prefix="/">
3         <file>images/b.jpg</file>
4     </qresource>
5 </RCC>
```

28.4 设计界面



C:\Users\Minwei\Projects\Qt\ReboundedBall\reboundedballdialog.ui:

```
1 <?xml version="1.0" encoding="UTF-8"?>
2 <ui version="4.0">
3 <class>ReboundedBallDialog</class>
4 <widget class="QDialog" name="ReboundedBallDialog">
5 <property name="geometry">
6 <rect>
7 <x>0</x>
8 <y>0</y>
9 <width>800</width>
10 <height>600</height>
11 </rect>
12 </property>
13 <property name="windowTitle">
14 <string>反弹小球</string>
15 </property>
16 <property name="styleSheet">
17 <string notr="true">border-image: url(/images/b.jpg);</string>
18 </property>
19 </widget>
20 <resources/>
21 <connections/>
22 </ui>
```

28.5 实现功能

C:\Users\Minwei\Projects\Qt\ReboundedBall\reboundedballdialog.h:

```
1 #ifndef REBOUNDEDBALLDIALOG_H
2 #define REBOUNDEDBALLDIALOG_H
3
4 #include <QDialog>
5
6 QT_BEGIN_NAMESPACE
7 namespace Ui { class ReboundedBallDialog; }
8 QT_END_NAMESPACE
9
10 class ReboundedBallDialog : public QDialog
11 {
12     Q_OBJECT
13
14 public:
15     ReboundedBallDialog(QWidget *parent = nullptr);
16     ~ReboundedBallDialog();
17
18 protected:
19     void paintEvent(QPaintEvent*);
20     void timerEvent(QTimerEvent*);
21
22 private:
23     Ui::ReboundedBallDialog *ui;
24     int m_dia;
25     QPoint m_pos, m_off;
```

```

26     int m_timer;
27 };
28
29 #endif // REBOUNDEDBALLDIALOG_H

```

C:\Users\Minwei\Projects\Qt\ReboundedBall\reboundedballdialog.cpp:

```

1  #include <algorithm>
2  using namespace std;
3
4  #include <QPainter>
5
6  #include "reboundedballdialog.h"
7  #include "ui_reboundedballdialog.h"
8
9  ReboundedBallDialog::ReboundedBallDialog(QWidget *parent)
10     : QDialog(parent)
11     , ui(new Ui::ReboundedBallDialog)
12     , m_dia(50)
13     , m_off(5, 5)
14  {
15     ui->setupUi(this);
16
17     m_pos.setX((width() - m_dia) / 2);
18     m_pos.setY((height() - m_dia) / 2);
19
20     m_timer = startTimer(10);
21  }
22
23  ReboundedBallDialog::~ReboundedBallDialog()
24  {
25     killTimer(m_timer);
26
27     delete ui;
28  }
29
30  void ReboundedBallDialog::paintEvent(QPaintEvent*)
31  {
32     QPainter painter(this);
33     painter.setRenderHint(QPainter::Antialiasing);
34
35     QPen pen(Qt::red);
36     painter.setPen(pen);
37     QRadialGradient brush(m_pos + QPoint(m_dia/3, m_dia/3), m_dia*2/3);
38     brush.setColorAt(0,Qt::white);
39     brush.setColorAt(1,Qt::red);
40     brush.setSpread(QGradient::PadSpread);
41     painter.setBrush(brush);
42
43     painter.drawEllipse(QRect(m_pos, m_pos + QPoint(m_dia, m_dia)));
44  }
45
46  void ReboundedBallDialog::timerEvent(QTimerEvent*)
47  {
48     m_pos += m_off;

```

```
49
50     int x = width() - m_dia, Y = height() - m_dia;
51
52     if (m_pos.x() < 0 || X < m_pos.x())
53     {
54         m_pos.setX(min(max(0, m_pos.x()), X));
55         m_off.setX(-m_off.x());
56     }
57
58     if (m_pos.y() < 0 || Y < m_pos.y())
59     {
60         m_pos.setY(min(max(0, m_pos.y()), Y));
61         m_off.setY(-m_off.y());
62     }
63
64     update();
65 }
```

28.6 测试验证

运行效果如图所示:

