

53 QStandardItemModel 案例功能

第二步，实现具体功能。

53.1 准备数据文件

C:\Users\Minwei\Projects\Qt\Table\debug\table1.csv:

```
1 学号,姓名,出生日期,性别,电话
2 1001,张飞,1996-09-12,男,13512345678
3 1002,赵云,1999-12-23,男,13487654321
4 1003,关羽,2000-11-11,男,13613267438
5 1004,马超,1998-05-20,男,13342740871
6 1005,黄忠,1997-07-15,男,13026539567
```

53.2 实现“打开”功能

C:\Users\Minwei\Projects\Qt\Table\tablewindow.h:

```
1 ...
2 #include <QStandardItemModel>
3 ...
4 class Tablewindow : public QMainWindow
5 {
6     ...
7 private:
8     void initModel(QStringList const& strings);
9     ...
10    QStandardItemModel* m_model;
11    ...
12 };
13 ...
```

C:\Users\Minwei\Projects\Qt\Table\tablewindow.cpp:

```
1 ...
2 #include <QFileDialog>
3 #include <QTextStream>
4 ...
5 Tablewindow::Tablewindow(QWidget *parent)
6     : QMainWindow(parent)
7     ...
8     , m_model(new QStandardItemModel(this))
9     ...
10 {
11     ...
12     ui->m_table->setModel(m_model);
13     ...
14 }
15 ...
16 void Tablewindow::on_m_actOpen_triggered()
17 {
18     QString path = QFileDialog::getOpenFileName(this,
```

```

19     "打开", QApplication::applicationDirPath(),
20     "逗号分隔值文件 (*.csv);;所有文件 (*.*)");
21     if (path.isEmpty())
22         return;
23
24     QFile file(path);
25     if (!file.open(QIODevice::ReadOnly | QIODevice::Text))
26         return;
27
28     QTextStream stream(&file);
29     stream.setCodec("utf-8");
30
31     QStringList strings;
32     while (!stream.atEnd())
33         strings.append(stream.readLine());
34
35     file.close();
36     initModel(strings);
37     m_labCurFile->setText("当前文件: " + path);
38 }
39 ...
40 void Tablewindow::initModel(QStringList const& strings)
41 {
42     m_model->clear();
43
44     m_model->setHorizontalHeaderLabels(strings.at(0).split(
45         QRegExp("\\\\,+"), QString::SkipEmptyParts));
46
47     int rowCount = strings.count() - 1;
48     for (int row = 0; row < rowCount; ++row)
49     {
50         QStringList values = strings.at(row + 1).split(
51             QRegExp("\\\\,+"), QString::SkipEmptyParts);
52
53         int columnCount = values.size();
54         for (int column = 0; column < columnCount; ++column)
55             m_model->setItem(row, column, new
QStandardItem(values.at(column)));
56     }
57 }
58 ...

```

53.3 实现“保存”功能

C:\Users\Minwei\Projects\Qt\Table\tablewindow.cpp:

```

1     ...
2 void Tablewindow::on_m_actSave_triggered()
3 {
4     QString path = QFileDialog::getSaveFileName(this,
5         "保存", QApplication::applicationDirPath(),
6         "逗号分隔值文件 (*.csv);;所有文件 (*.*)");
7     if (path.isEmpty())
8         return;
9

```

```

10     QFile file(path);
11     if(!(file.open(QIODevice::ReadWrite | QIODevice::Text |
12         QIODevice::Truncate)))
13         return;
14
15     QTextStream stream(&file);
16     stream.setCodec("utf-8");
17
18     int columnCount = m_model->columnCount();
19     for (int column = 0; column < columnCount; ++column)
20         stream << m_model->horizontalHeaderItem(column)->text() <<
21             (column == columnCount - 1 ? "\n" : ",");
22
23     int rowCount = m_model->rowCount();
24     for (int row = 0; row < rowCount; ++row)
25         for (int column = 0; column < columnCount; ++column)
26             stream << m_model->item(row, column)->text() <<
27                 (column == columnCount - 1 ? "\n" : ",");
28
29     file.close();
30 }
31 ...

```

53.4 实现“预览”功能

C:\Users\Minwei\Projects\Qt\Table\tablewindow.cpp:

```

1     ...
2     void Tablewindow::on_m_actPreview_triggered()
3     {
4         ui->m_edit->clear();
5
6         QString text;
7         int columnCount = m_model->columnCount();
8         for (int column = 0; column < columnCount; ++column)
9             text += m_model->horizontalHeaderItem(column)->text() +
10                 (column == columnCount - 1 ? "" : ",");
11         ui->m_edit->appendPlainText(text);
12
13         int rowCount = m_model->rowCount();
14         for (int row = 0; row < rowCount; ++row)
15         {
16             QString text;
17             for (int column = 0; column < columnCount; ++column)
18                 text += m_model->item(row, column)->text() +
19                     (column == columnCount - 1 ? "" : ",");
20             ui->m_edit->appendPlainText(text);
21         }
22     }
23     ...

```

53.5 实现“添加”功能

C:\Users\Minwei\Projects\Qt\Table\tablewindow.cpp:

```
1  ...
2  void Tablewindow::on_m_actAppend_triggered()
3  {
4      int columnCount = m_model->columnCount();
5      if (!columnCount)
6          return;
7
8      QList<QStandardItem*> items;
9      for (int column = 0; column < columnCount; ++column)
10         items << new QStandardItem;
11     m_model->insertRow(m_model->rowCount(), items);
12 }
13 ...
```

53.6 实现“插入”功能

C:\Users\Minwei\Projects\Qt\Table\tablewindow.h:

```
1  ...
2  #include <QItemSelectionMode>
3  ...
4  class Tablewindow : public QMainWindow
5  {
6      ...
7  private:
8      ...
9      QItemSelectionMode* m_selection;
10     ...
11 };
12 ...
```

C:\Users\Minwei\Projects\Qt\Table\tablewindow.cpp:

```
1  ...
2  Tablewindow::Tablewindow(QWidget *parent)
3      : QMainWindow(parent)
4      ...
5      , m_selection(new QItemSelectionMode(m_model))
6      ...
7  {
8      ...
9      ui->m_table->setSelectionMode(m_selection);
10     ...
11 }
12 ...
13 void Tablewindow::on_m_actAppend_triggered()
14 {
15     ...
16     m_selection->clearSelection();
17     m_selection->setCurrentIndex(m_model->index(m_model->rowCount() - 1, 0),
```

```

18     QTableWidgetItem::select);
19     ...
20 }
21 ...
22 void Tablewindow::on_m_actInsert_triggered()
23 {
24     int columnCount = m_model->columnCount();
25     if (!columnCount)
26         return;
27
28     QList<QStandardItem*> items;
29     for (int column = 0; column < columnCount; ++column)
30         items << new QStandardItem;
31
32     QModelIndex current = m_selection->currentIndex();
33     m_model->insertRow(current.row(), items);
34
35     m_selection->clearSelection();
36     m_selection->setCurrentIndex(current, QTableWidgetItem::select);
37 }
38 ...
39 void Tablewindow::initModel(QStringList const& strings)
40 {
41     ...
42     m_selection->setCurrentIndex(m_model->index(0, 0),
43         QTableWidgetItem::select);
44     ...
45 }
46 ...

```

53.7 实现“删除”功能

C:\Users\Minwei\Projects\Qt\Table\tablewindow.cpp:

```

1     ...
2 void Tablewindow::on_m_actDelete_triggered()
3 {
4     QModelIndex current = m_selection->currentIndex();
5     QModelIndex above = m_model->index(current.row() - 1, current.column());
6     bool last = current.row() == m_model->rowCount() - 1;
7
8     m_model->removeRow(current.row());
9     m_selection->setCurrentIndex(last ? above : current,
10         QTableWidgetItem::select);
11 }
12 ...

```

53.8 实现“靠左”、“居中”、“靠右”和“加粗”功能

C:\Users\Minwei\Projects\Qt\Table\tablewindow.cpp:

```

1     ...
2 void Tablewindow::on_m_actLeft_triggered()
3 {

```

```

4     for (QModelIndex const& index : m_selection->selectedIndexes())
5         m_model->itemFromIndex(index)->setTextAlignment(
6             Qt::AlignLeft | Qt::AlignVCenter);
7     }
8
9     void Tablewindow::on_m_actCenter_triggered()
10    {
11        for (QModelIndex const& index : m_selection->selectedIndexes())
12            m_model->itemFromIndex(index)->setTextAlignment(
13                Qt::AlignHCenter | Qt::AlignVCenter);
14    }
15
16    void Tablewindow::on_m_actRight_triggered()
17    {
18        for (QModelIndex const& index : m_selection->selectedIndexes())
19            m_model->itemFromIndex(index)->setTextAlignment(
20                Qt::AlignRight | Qt::AlignVCenter);
21    }
22
23    void Tablewindow::on_m_actBold_triggered(bool checked)
24    {
25        for (QModelIndex const& index : m_selection->selectedIndexes())
26        {
27            QStandardItem* item = m_model->itemFromIndex(index);
28            QFont font = item->font();
29            font.setBold(checked);
30            item->setFont(font);
31        }
32    }
33    ...

```

53.9 实现状态栏显示单元格信息功能

C:\Users\Minwei\Projects\Qt\Table\tablewindow.h:

```

1     ...
2     class Tablewindow : public QMainWindow
3     {
4         ...
5     private slots:
6         ...
7         void on_m_selection_currentChanged(
8             QModelIndex const& current, QModelIndex const& previous);
9         ...
10    };
11    ...

```

C:\Users\Minwei\Projects\Qt\Table\tablewindow.cpp:

```

1     ...
2     Tablewindow::Tablewindow(QWidget *parent)
3         : QMainWindow(parent)
4         ...
5     {

```

```

6     ...
7     connect(m_selection, SIGNAL(currentChanged(QModelIndex, QModelIndex)),
8             this, SLOT(on_m_selection_currentChanged(QModelIndex, QModelIndex)));
9     ...
10  }
11  ...
12  void Tablewindow::on_m_selection_currentChanged(
13      QModelIndex const& current, QModelIndex const& previous)
14  {
15      Q_UNUSED(previous)
16
17      if (!current.isValid())
18          return;
19
20      m_labCellPos->setText(QString(" 单元格位置: 第%1行, 第%2列")
21                             .arg(current.row() + 1).arg(current.column() + 1));
22
23      QStandardItem* item = m_model->itemFromIndex(current);
24      m_labCellText->setText(" 单元格内容: " + item->text());
25
26      ui->m_actBold->setChecked(item->font().bold());
27  }
28  ...

```

53.10 测试验证

运行效果如图所示:

