

ID	...	Creation date	26 June 2009
Platform	S60 3rd Edition, S60 5th Edition	Tested on devices	S60 Emulator
Category	Qt for Symbian	Subcategory	UI

Keywords (APIs, classes, methods, functions): QStandardItemModel, QDataWidgetMapper

Overview

This will show you the mapping of the StandardItemModel to DataWidgetMapper and directly to QLineEdit.

QDataWidgetMapper can be used to create data-aware widgets by mapping them to sections of an item model. A section is a column of a model if the orientation is horizontal (the default), otherwise a row.

Preconditions

- Download and Install latest version [Qt for Symbian - Installation packages](#) which has links on how to install the latest version
- Download Qt creator IDE: [QtIDEv4.5](#)

Source Code

Header File

```
#ifndef MODEL_H
#define MODEL_H

#include <QtGui/QWidget>
#include<QLabel>
#include<QLineEdit>
#include<QPushButton>
#include<QStandardItemModel>
#include<QStringListModel>
#include<QDataWidgetMapper>
#include<QGridLayout>
#include<QStandardItem>
#include<QStringList>
class model : public QWidget
{
    Q_OBJECT

public:
```

Mapping_of_StandardItemModel_via_DataWidgetMapper_in_Qt_for_Symbian

```
model(QWidget *parent = 0);
~model();

private slots:
    void updateButtons(int row);

private:
    void virModel();

    QLabel *nameLabel;
    QLineEdit *nameEdit;
    QPushButton *nextButton;
    QPushButton *previousButton;
    QStandardItemModel *modelversion;
    QStringListModel *typeModel;
    QDataWidgetMapper *mapper;
};

#endif // MODEL_H
```

Source File

```
#include "model.h"
model::model(QWidget *parent)
    : QWidget(parent)
{
    virModel();

    nameLabel = new QLabel(tr("Name:"), this);
    nameEdit = new QLineEdit(this);
    nameLabel->setBuddy(nameEdit);
    nextButton = new QPushButton(tr("&Next"), this);
    previousButton = new QPushButton(tr("&Previous"), this);
    mapper = new QDataWidgetMapper(this);
    mapper->setModel(modelversion); //mapping of the dataModel is Done
    mapper->addMapping(nameEdit, 0);
    connect(previousButton, SIGNAL(clicked()), mapper, SLOT(toPrevious()));
    connect(nextButton, SIGNAL(clicked()),
            mapper, SLOT(toNext()));
    connect(mapper, SIGNAL(currentIndexChanged(int)),
            this, SLOT(updateButtons(int)));

    QGridLayout *layout = new QGridLayout(this); //Widget Layout is set
    layout->addWidget(nameLabel, 0, 0, 1, 1);
    layout->addWidget(nameEdit, 0, 1, 1, 1);
    layout->addWidget(previousButton, 0, 2, 1, 1);
    layout->addWidget(nextButton, 1, 2, 1, 1);
    setLayout(layout);
    setWindowTitle(tr("Widget Mapper"));
    mapper->toFirst();
}

model::~~model()
{
    // No need to delete any object that has got a parent which is properly deleted.
}

void model::virModel() //Whole the stringlist items are added to standardItemModel
{
    modelversion = new QStandardItemModel(5, 1, this);
```

Mapping_of_StandardItemModel_via_DataWidgetMapper_in_Qt_for_Symbian

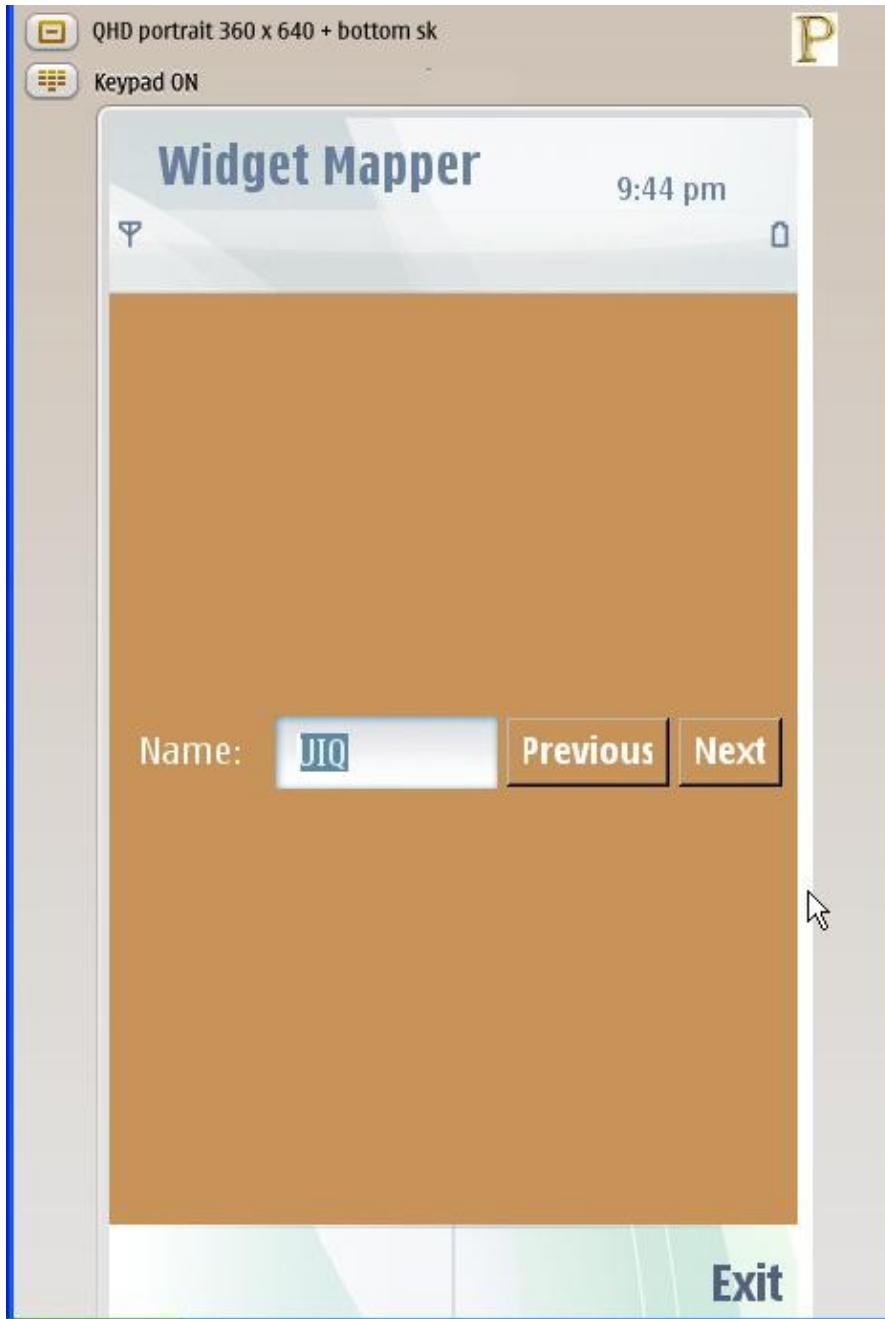
```
QStringList names;
names << "symbian" << "UIQ" << "ANDROID" << "NOKIA" << "APPLE";

for (int row = 0; row < 5; ++row)
{
    QStandardItem *item = new QStandardItem(names[row]);
    modelversion->setItem(row, 0, item);
}

}

void model::updateButtons(int row) //Module Execute when their is updatation in QLineEdit
{
    previousButton->setEnabled(row > 0);
    nextButton->setEnabled(row < modelversion->rowCount() - 1);
}
```

Screenshot



Related Links

[Mapping signal via signalMapper](#)