

```
#include "addressbook.h"
#include "ui_addressbook.h"

AddressBook::AddressBook(QWidget *parent)
    : QWidget(parent), ui(new Ui::AddressBook)
{
    ui->setupUi(this);

    nameLine = new QLineEdit;
    nameLine = ui->nameLine;
    nameLine->setReadOnly(true);

    addressText = new QTextEdit;
    addressText = ui->addressText;
    addressText->setReadOnly(true);

    addButton = new QPushButton;
    addButton = ui->addButton;

    submitButton = new QPushButton;
    submitButton = ui->submitButton;
    submitButton->hide();

    cancelButton = new QPushButton;
    cancelButton = ui->cancelButton;
    cancelButton->hide();

    nextButton = new QPushButton;
    nextButton = ui->nextButton;
    nextButton->setEnabled(false);

    previousButton = new QPushButton;
    previousButton = ui->previousButton;
    nextButton->setEnabled(false);

    connect(addButton, SIGNAL(clicked()), this,
            SLOT(addContact()));
    connect(submitButton, SIGNAL(clicked()), this,
            SLOT(submitContact()));
    connect(cancelButton, SIGNAL(clicked()), this,
            SLOT(cancel()));
    connect(nextButton, SIGNAL(clicked()), this,
            SLOT(next()));
    connect(previousButton, SIGNAL(clicked()), this,
            SLOT(previous()));

    setWindowTitle(tr("Simple Address Book"));
}

AddressBook::~AddressBook()
{
    delete ui;
}

void AddressBook::addContact()
{
    oldName = nameLine->text();
    oldAddress = addressText->toPlainText();

    nameLine->clear();
    addressText->clear();
}
```

```
nameLine->setReadOnly(false);
nameLine->setFocus(Qt::OtherFocusReason);
addressText->setReadOnly(false);

addButton->setEnabled(false);
nextButton->setEnabled(false);
previousButton->setEnabled(false);
submitButton->show();
cancelButton->show();
}

void AddressBook::submitContact()
{
    QString name = nameLine->text();
    QString address = addressText->toPlainText();

    if (name == "" || address == "") {
        QMessageBox::information(this, tr("Empty Field"),
            tr("Please enter a name and address.));
        return;
    }

    if (!contacts.contains(name)) {
        contacts.insert(name, address);
        QMessageBox::information(this, tr("Add Successful"),
            tr("\'%1\' has been added to your address book.")).arg(name));
        return;
    } else {
        QMessageBox::information(this, tr("Add Unsuccessful"),
            tr("Sorry, \'%1\' is already in your address book.")).arg(name));
        return;
    }

    if (contacts.isEmpty()) {
        nameLine->clear();
        addressText->clear();
    }

    nameLine->setReadOnly(true);
    addressText->setReadOnly(true);
    addButton->setEnabled(true);

    int number = contacts.size();
    nextButton->setEnabled(number > 1);
    previousButton->setEnabled(number > 1);
    submitButton->hide();
    cancelButton->hide();
}

void AddressBook::cancel()
{
    nameLine->setText(oldName);
    nameLine->setReadOnly(true);

    addressText->setText(oldAddress);
    addressText->setReadOnly(true);
    addButton->setEnabled(true);

    int number = contacts.size();
    nextButton->setEnabled(number > 1);
    previousButton->setEnabled(number > 1);
}
```

```
        submitButton->hide();
        cancelButton->hide();
    }

    void AddressBook::next()
    {
        QString name = nameLine->text();
        QMap<QString, QString>::iterator i = contacts.find(name);

        if (i != contacts.end())
            i++;
        if (i == contacts.end())
            i = contacts.begin();

        nameLine->setText(i.key());
        addressText->setText(i.value());
    }

    void AddressBook::previous()
    {
        QString name = nameLine->text();
        QMap<QString, QString>::iterator i = contacts.find(name);

        if (i == contacts.end()) {
            nameLine->clear();
            addressText->clear();
            return;
        }

        if (i == contacts.begin())
            i = contacts.end();

        i--;
        nameLine->setText(i.key());
        addressText->setText(i.value());
    }
}
```