Летняя школа по параллельному программированию 2016

"Введение в разработку переносимых приложений с GUI на основе фреймворка Qt"

Максим Александрович Городничев

ИВМиМГ СО РАН, НГУ, НГТУ

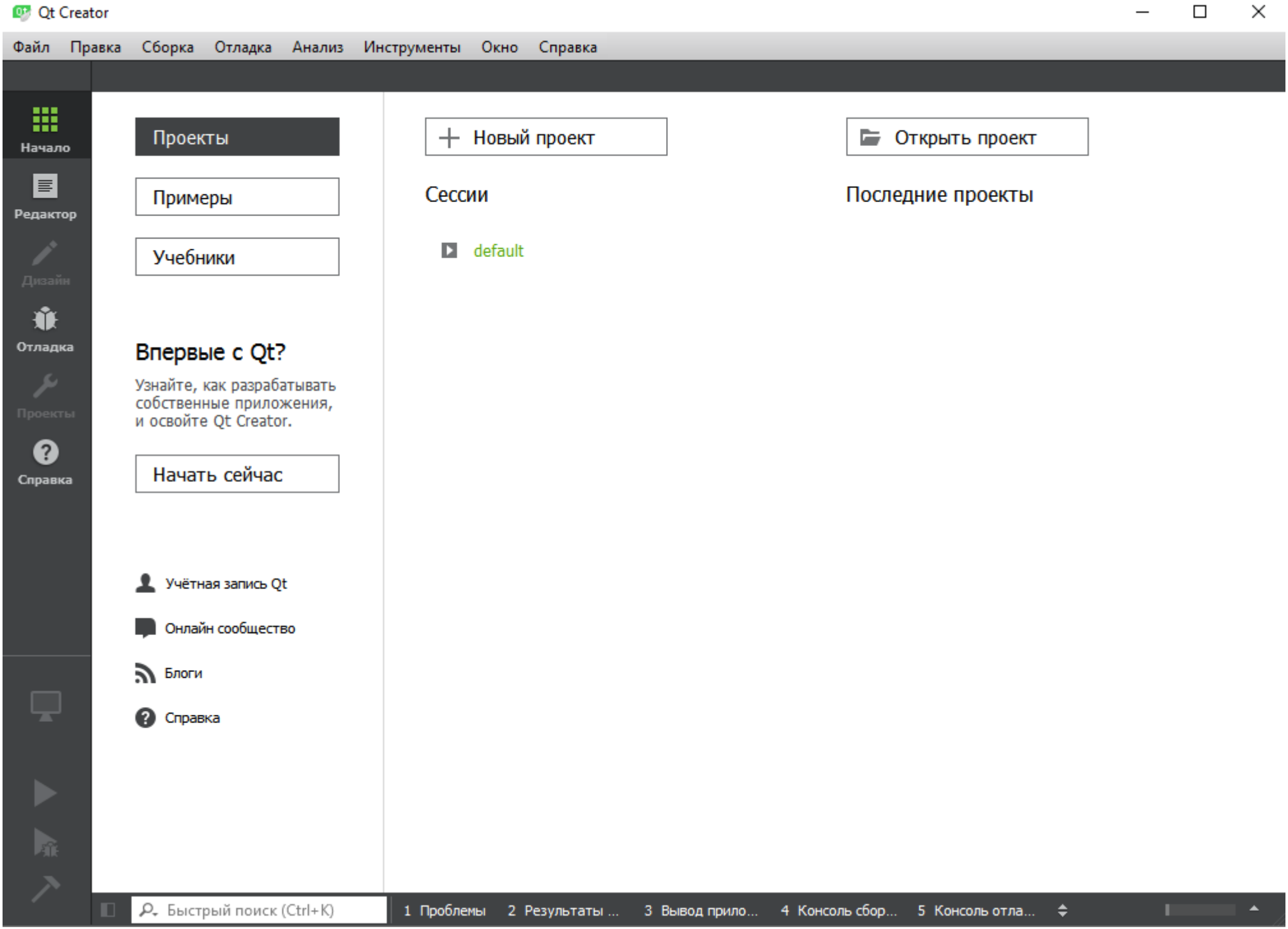
4-15.07.2016

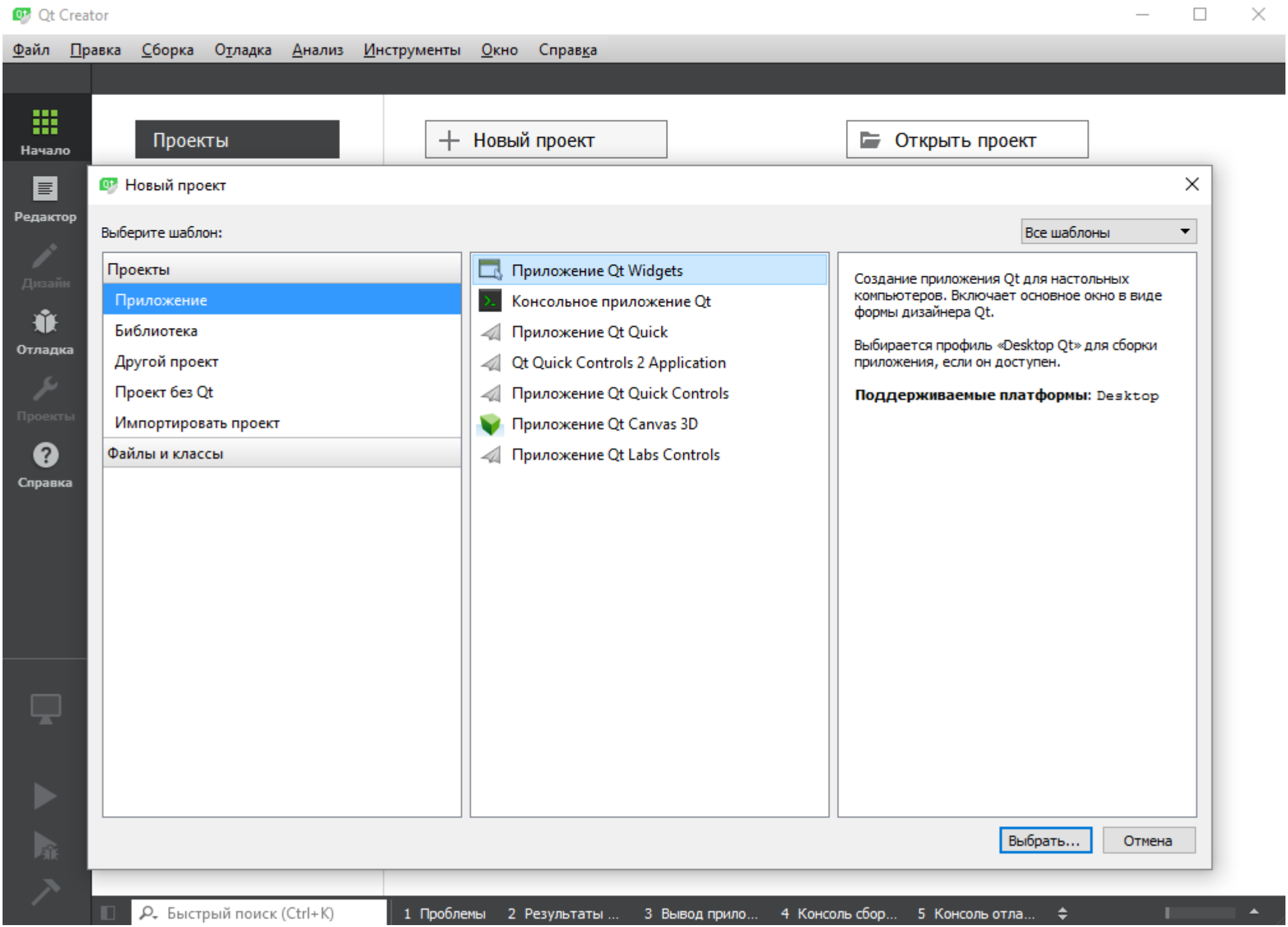
Qt [cute] – это

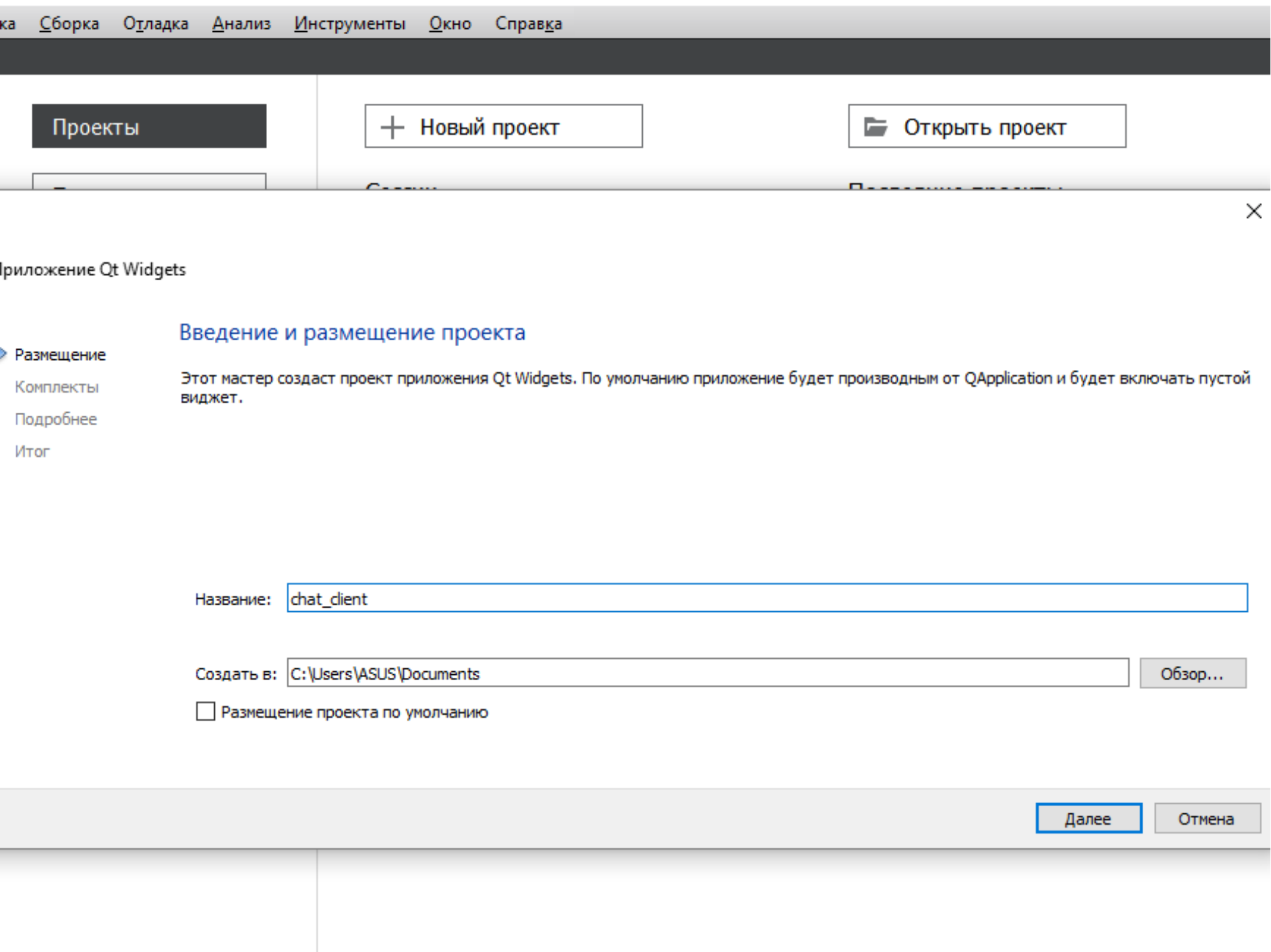
программная платформа для создания переносимых приложений с графическим пользовательским интерфейсом:

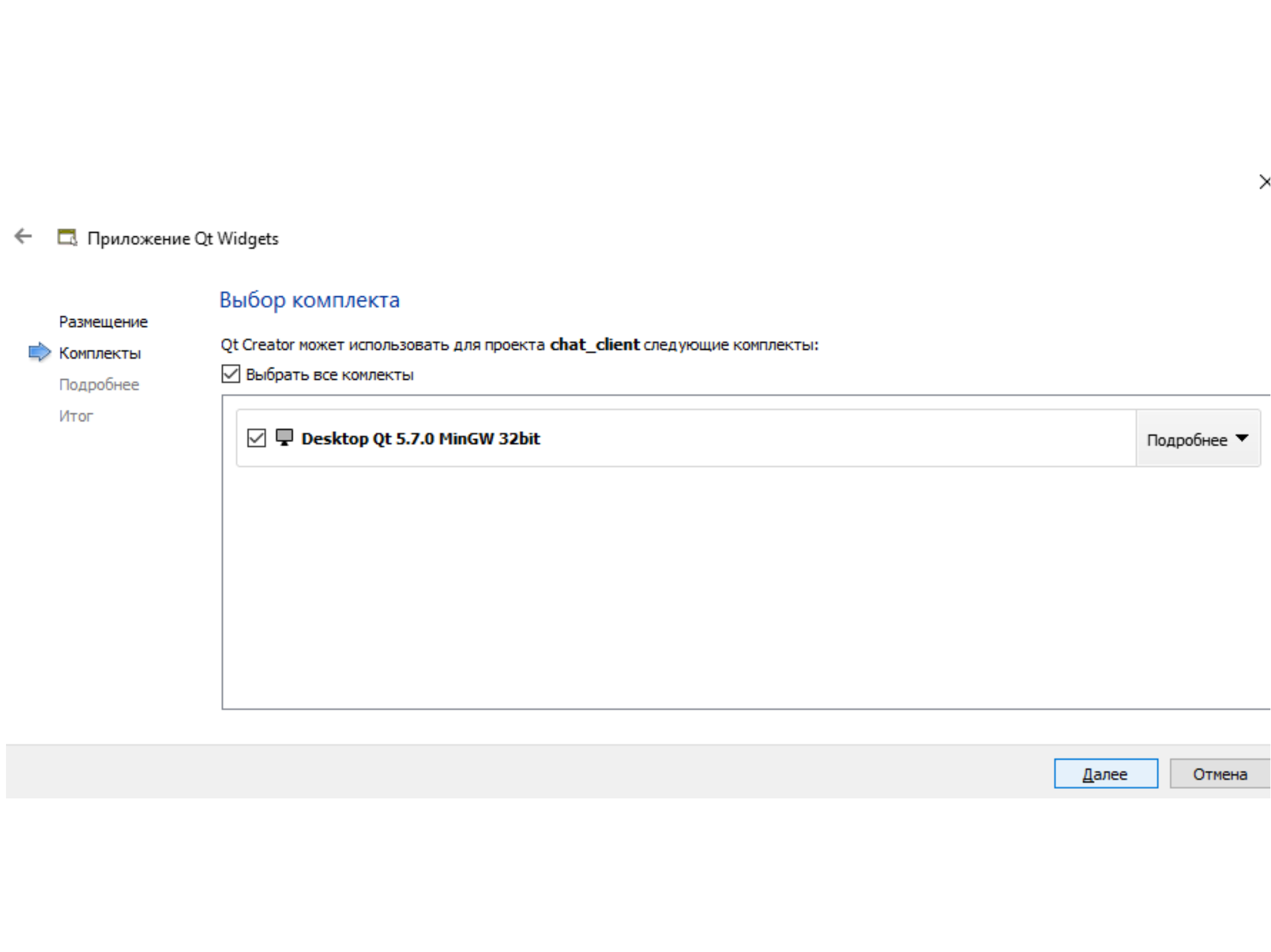
- модель приложения с сигналами и слотами,
- обширная библиотека классов C++,
- визуальный инструмент для рисования QUI (QtDesigner),
- исходные коды и документация,
- система сборки qmake,
- IDE: Qt Creator,
- QtLinquist,
- Qt Quick (QML),
- ...

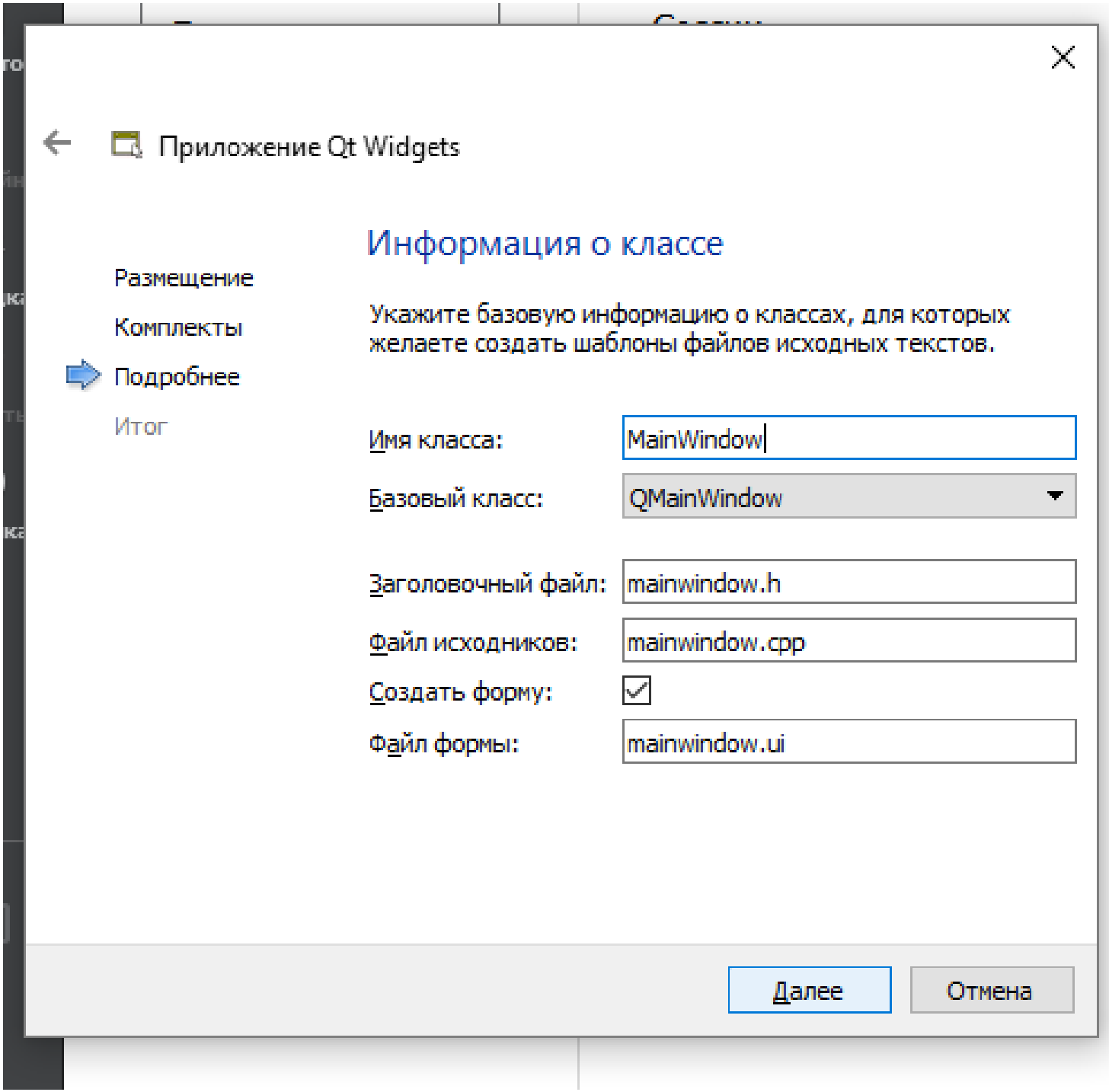
https://www.qt.io/

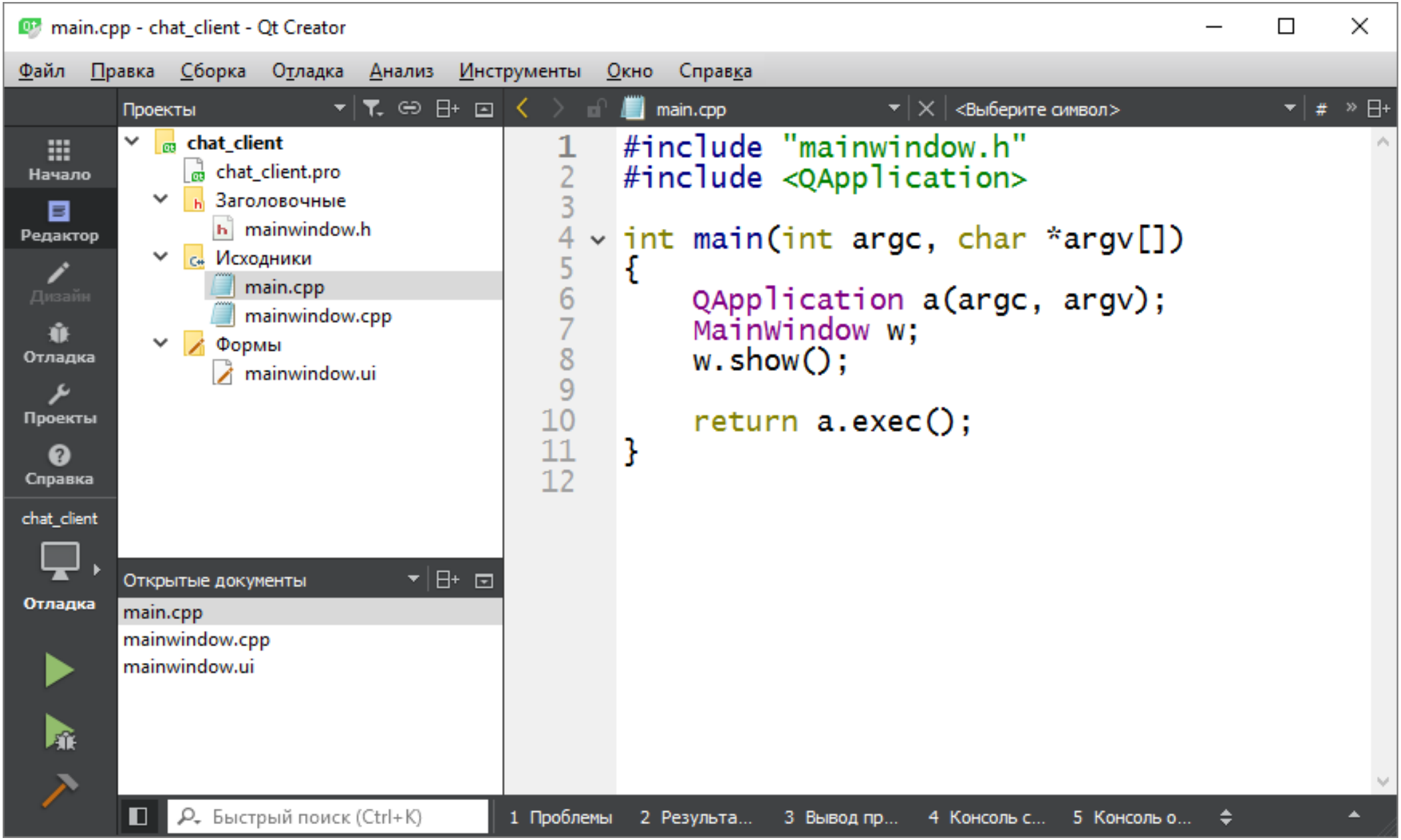

Qt Creator
Файл Правка Сборка Отладка Анализ Инструменты Окно Справка
Начало
Редактор
Дизайн
Отладка
Проекты
Справка
Проекты
Примеры
Учебники
Впервые с Qt?
Узнайте, как разрабатывать собственные приложения, и освойте Qt Creator.
Начать сейчас
Учётная запись Qt
Онлайн сообщество
Блоги
Справка
Новый проект
Открыть проект
Сессии
default
Последние проекты
Быстрый поиск (Ctrl+K)
1 Проблемы
2 Результаты ...
3 Вывод прило...
4 Консоль сбор...
5 Консоль отла...

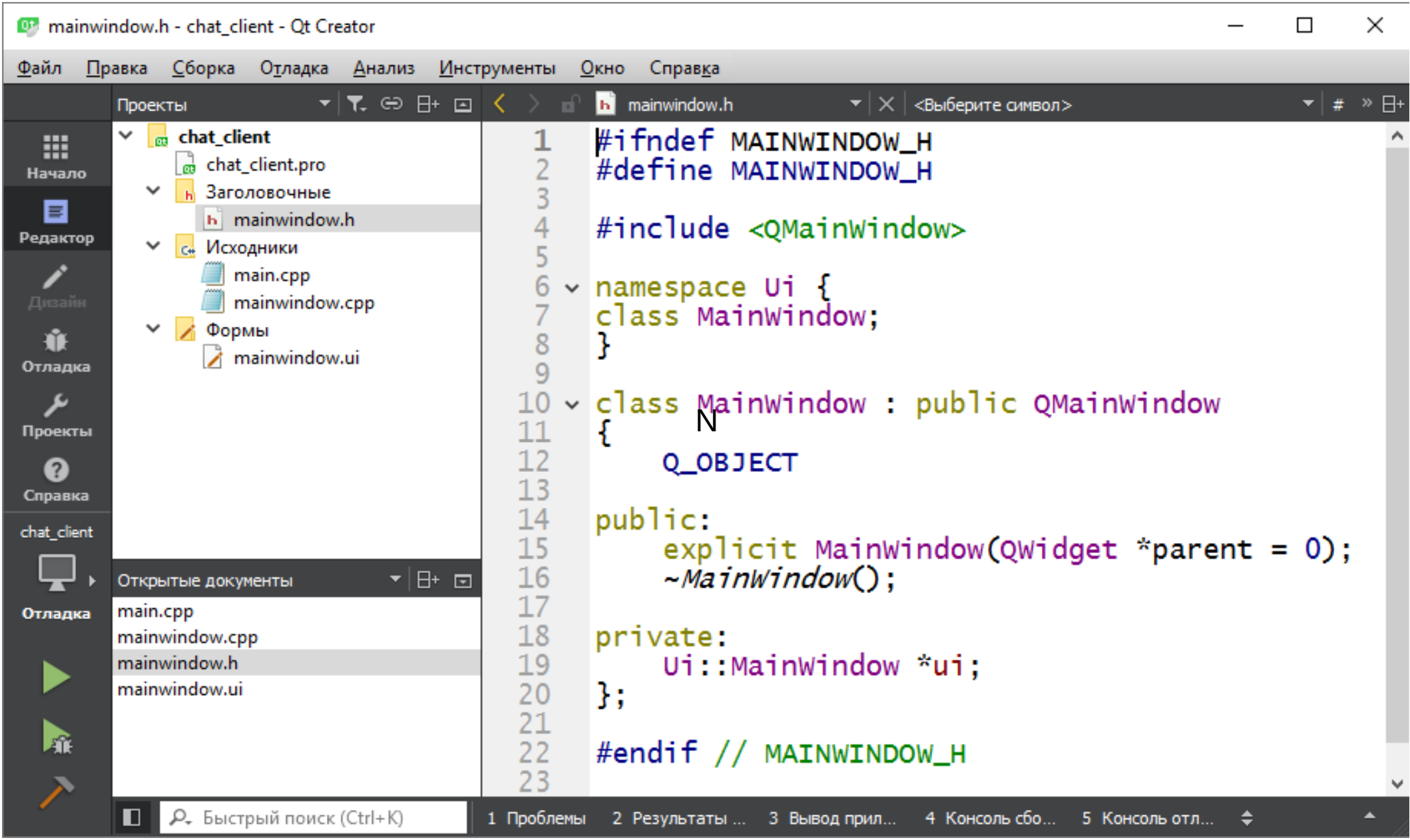

Qt Creator
Файл Правка Сборка Отладка Анализ Инструменты Окно Справка
Начало
Редактор
Дизайн
Отладка
Проекты
Справка
Проекты
Новый проект
Открыть проект
Новый проект
Выберите шаблон:
Все шаблоны
Проекты
Приложение
Библиотека
Другой проект
Проект без Qt
Импортировать проект
Файлы и классы
Приложение Qt Widgets
Консольное приложение Qt
Приложение Qt Quick
Qt Quick Controls 2 Application
Приложение Qt Quick Controls
Приложение Qt Canvas 3D
Приложение Qt Labs Controls
Создание приложения Qt для настольных компьютеров. Включает основное окно в виде формы дизайнера Qt.
Выбирается профиль «Desktop Qt» для сборки приложения, если он доступен.
Поддерживаемые платформы: Desktop
Выбрать...
Отмена
Быстрый поиск (Ctrl+K)
1 Проблемы
2 Результаты ...
3 Вывод прило...
4 Консоль сбор...
5 Консоль отла...


ка Сборка Отладка Анализ Инструменты Окно Справка
Проекты
Новый проект
Открыть проект
риложение Qt Widgets
Размещение
Комплекты
Подробнее
Итог
Введение и размещение проекта
Этот мастер создаст проект приложения Qt Widgets. По умолчанию приложение будет производным от QApplication и будет включать пустой виджет.
Название:
chat_client
Создать в:
C:\Users\ASUS\Documents
Обзор...
Размещение проекта по умолчанию
Далее
Отмена


Приложение Qt Widgets
Выбор комплекта
Размещение
Комплекты
Подробнее
Итог
Qt Creator может использовать для проекта chat_client следующие комплекты:
Выбрать все комплекты
Desktop Qt 5.7.0 MinGW 32bit
Подробнее
Далее
Отмена


Приложение Qt Widgets
Размещение
Комплекты
Подробнее
Итог
Информация о классе
Укажите базовую информацию о классах, для которых желаете создать шаблоны файлов исходных текстов.
Имя класса:
MainWindow
Базовый класс:
QMainWindow
Заголовочный файл:
mainwindow.h
Файл исходников:
mainwindow.cpp
Создать форму:
Файл формы:
mainwindow.ui
Далее
Отмена


main.cpp - chat_client - Qt Creator
Файл Правка Сборка Отладка Анализ Инструменты Окно Справка
Проекты
chat_client
chat_client.pro
Заголовочные
mainwindow.h
Исходники
main.cpp
mainwindow.cpp
Формы
mainwindow.ui
Начало
Редактор
Дизайн
Отладка
Проекты
Справка
chat_client
Отладка
Открытые документы
main.cpp
mainwindow.cpp
mainwindow.ui
main.cpp
<Выберите символ>
#include "mainwindow.h"
#include <QApplication>
int main(int argc, char *argv[])
{
QApplication a(argc, argv);
MainWindow w;
w.show();
return a.exec();
}
Быстрый поиск (Ctrl+K)
1 Проблемы
2 Результа...
3 Вывод пр...
4 Консоль с...
5 Консоль о...


mainwindow.h - chat_client - Qt Creator
Файл Правка Сборка Отладка Анализ Инструменты Окно Справка
Проекты
chat_client
chat_client.pro
Заголовочные
mainwindow.h
Исходники
main.cpp
mainwindow.cpp
Формы
mainwindow.ui
Начало
Редактор
Дизайн
Отладка
Проекты
Справка
chat_client
Отладка
Открытые документы
main.cpp
mainwindow.cpp
mainwindow.h
mainwindow.ui
mainwindow.h
<Выберите символ>
#ifndef MAINWINDOW_H
#define MAINWINDOW_H
#include <QMainWindow>
namespace Ui {
class MainWindow;
}
class MainWindow : public QMainWindow
{
Q_OBJECT
public:
explicit MainWindow(QWidget *parent = 0);
~MainWindow();
private:
Ui::MainWindow *ui;
};
#endif // MAINWINDOW_H
Быстрый поиск (Ctrl+K)
1 Проблемы
2 Результаты ...
3 Вывод прил...
4 Консоль сбо...
5 Консоль отл...

QMainWindow Class

The QMainWindow class provides a main application window. More...

Header:	#include <QMainWindow>
qmake:	QT += widgets
Inherits:	QWidget

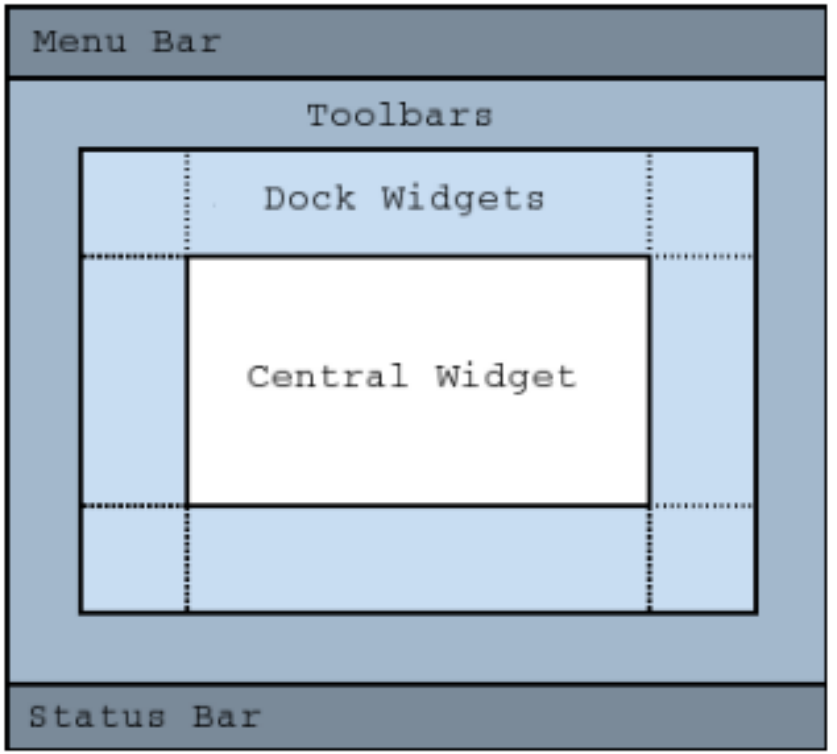
› List of all members, including inherited members

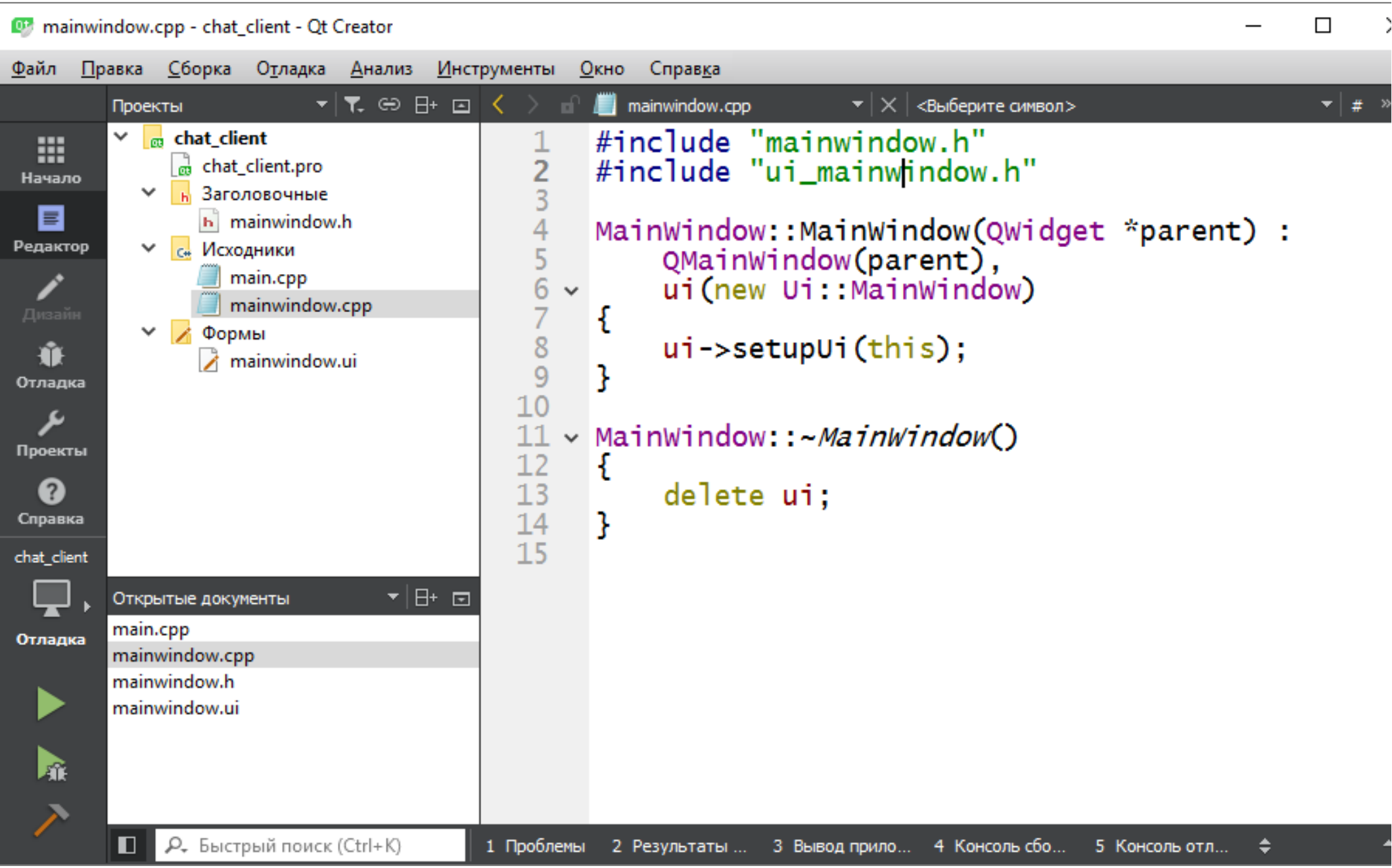
Public Types

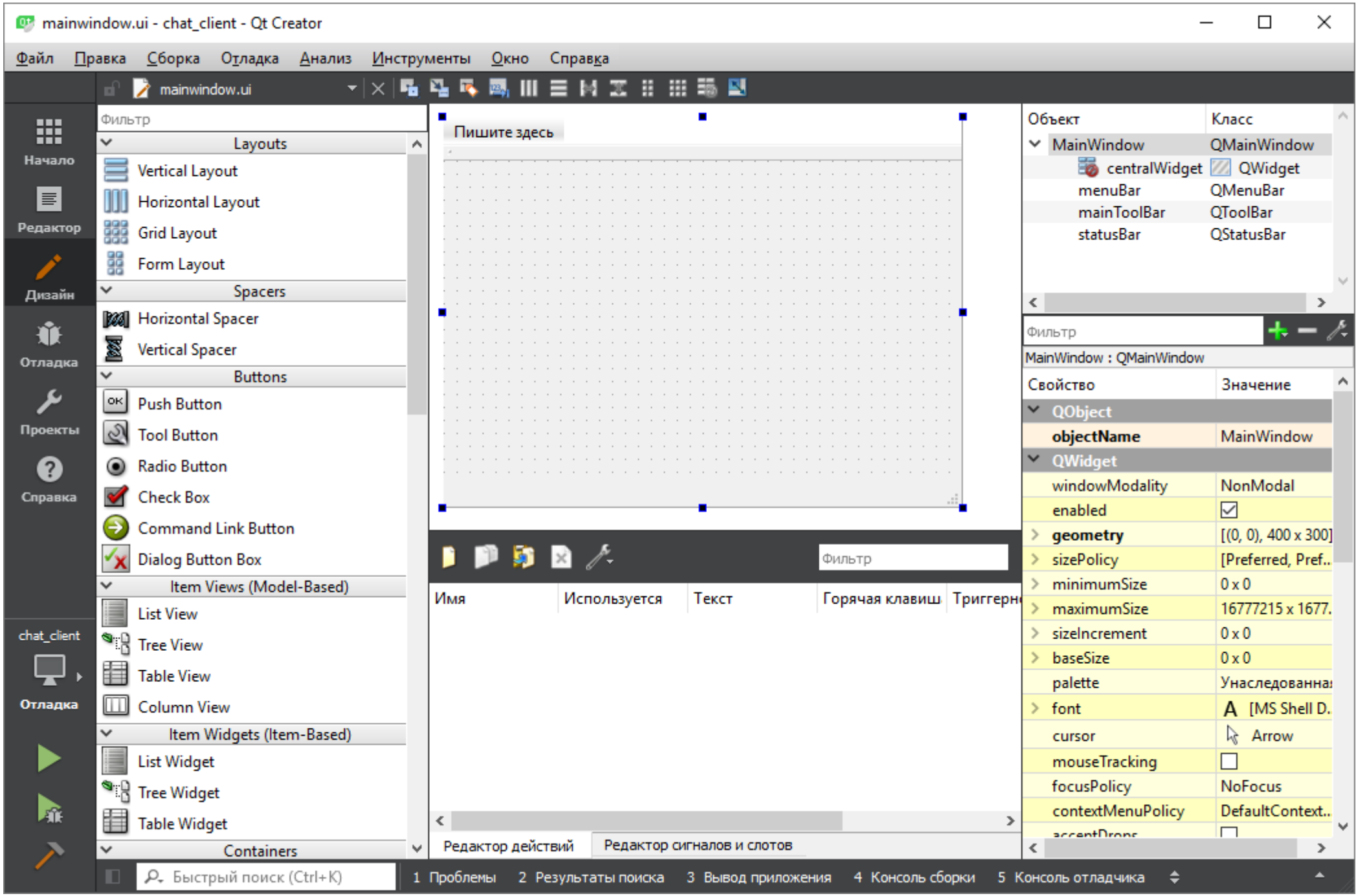
enum	DockOption { AnimatedDocks, AllowNestedDocks, AllowTabbedDocks, ForceTabbedDocks, VerticalTabs, Grou
flags	DockOptions

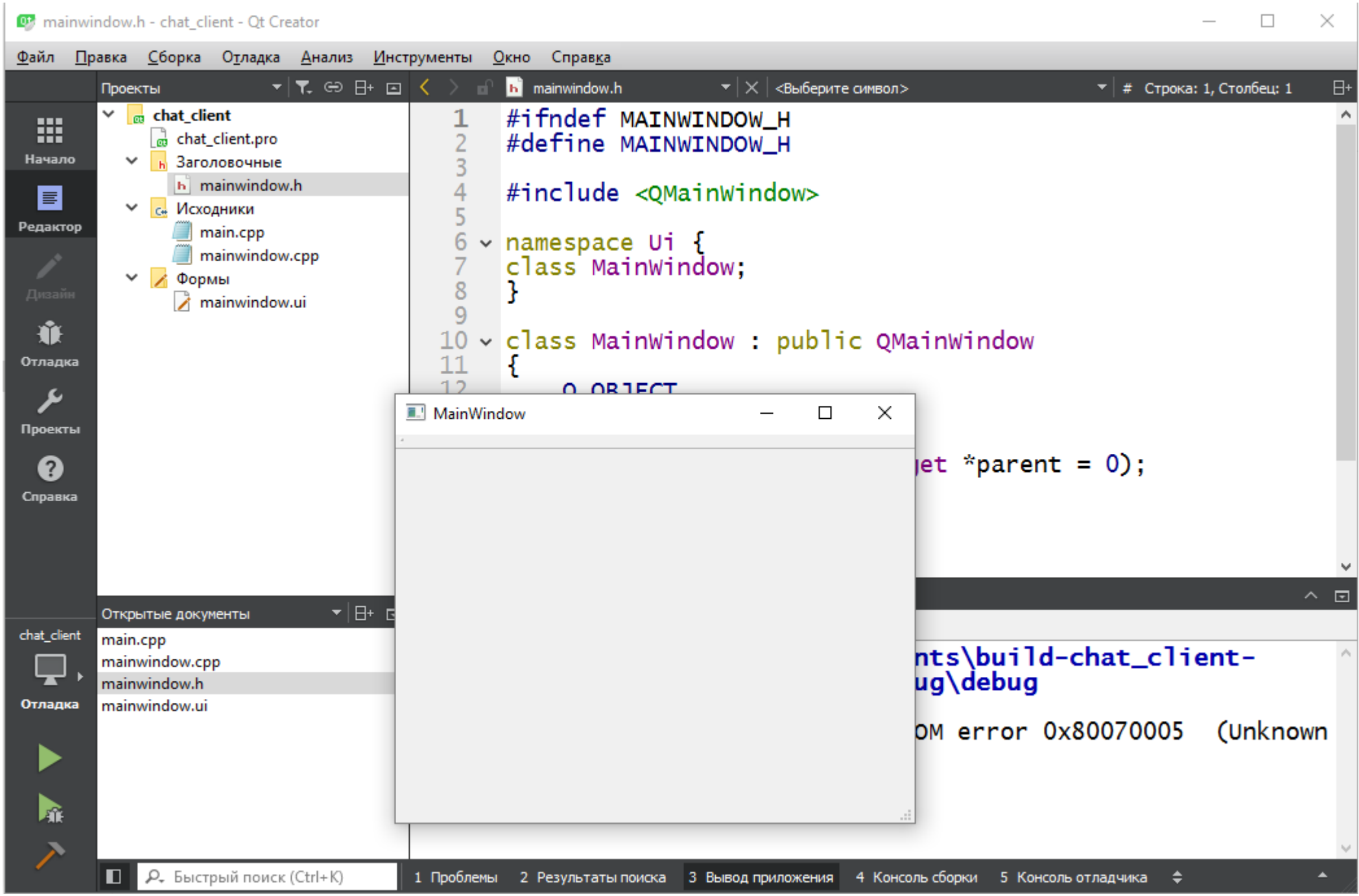
Qt Main Window Framework

A main window provides a framework for building an application's user interface. Qt has QMainWindow and its related classes for main window management. QMainWindow has its own layout to which you can add QToolBars, QDockWidgets, a QMenuBar, and a QStatusBar. The layout has a center area that can be occupied by any kind of widget. You can see an image of the layout below.

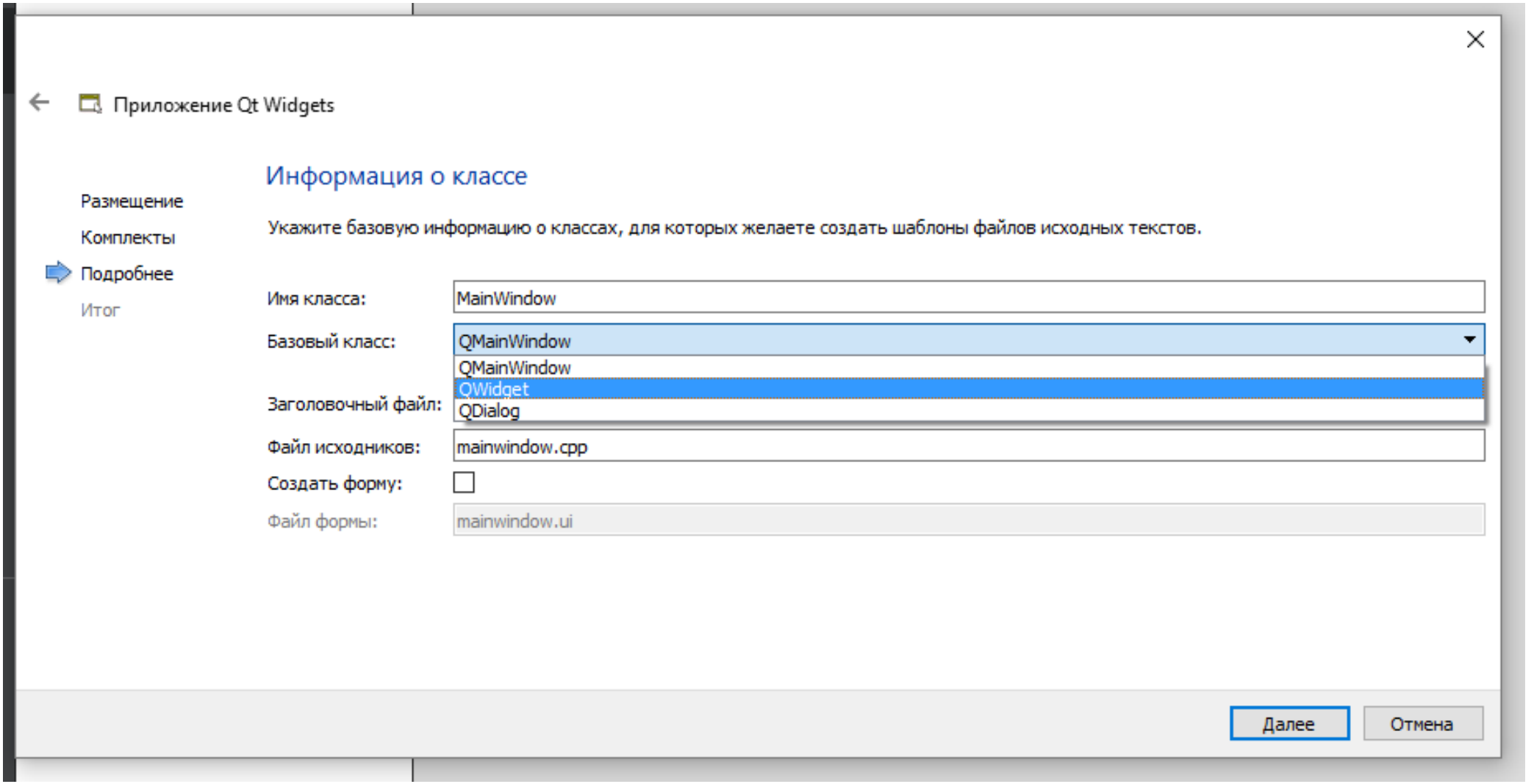



mainwindow.cpp - chat_client - Qt Creator
Файл Правка Сборка Отладка Анализ Инструменты Окно Справка
Проекты
chat_client
chat_client.pro
Заголовочные
mainwindow.h
Исходники
main.cpp
mainwindow.cpp
Формы
mainwindow.ui
Начало
Редактор
Дизайн
Отладка
Проекты
Справка
chat_client
Отладка
Открытые документы
main.cpp
mainwindow.cpp
mainwindow.h
mainwindow.ui
mainwindow.cpp
<Выберите символ>
#include "mainwindow.h"
#include "ui_mainwindow.h"
MainWindow::MainWindow(QWidget *parent) :
QMainWindow(parent),
ui(new Ui::MainWindow)
{
ui->setupUi(this);
}
MainWindow::~MainWindow()
{
delete ui;
}
Быстрый поиск (Ctrl+K)
1 Проблемы
2 Результаты ...
3 Вывод прило...
4 Консоль сбо...
5 Консоль отл...

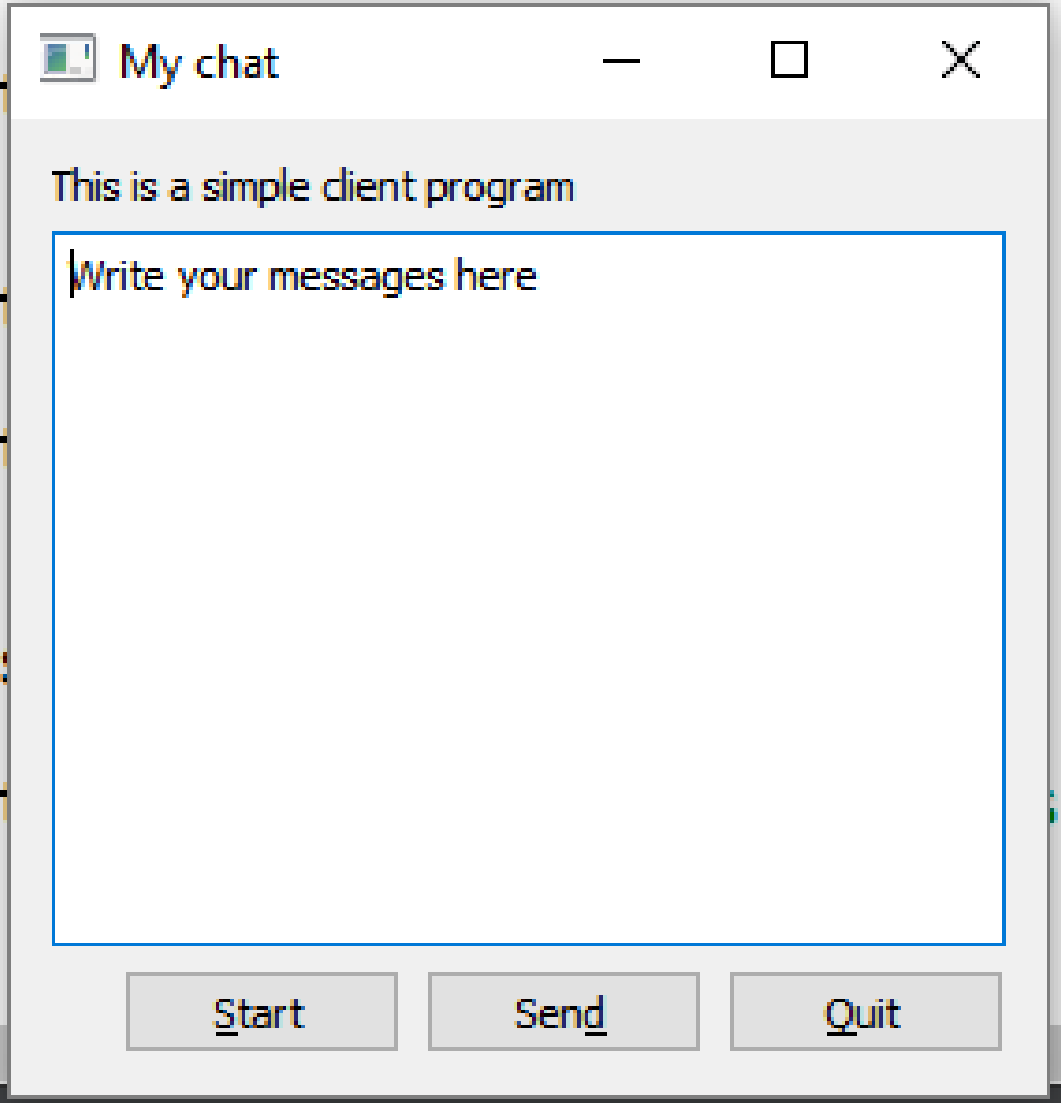

mainwindow.ui - chat_client - Qt Creator
Файл Правка Сборка Отладка Анализ Инструменты Окно Справка
mainwindow.ui
Начало
Редактор
Дизайн
Отладка
Проекты
Справка
chat_client
Отладка
Фильтр
Layouts
Vertical Layout
Horizontal Layout
Grid Layout
Form Layout
Spacers
Horizontal Spacer
Vertical Spacer
Buttons
Push Button
Tool Button
Radio Button
Check Box
Command Link Button
Dialog Button Box
Item Views (Model-Based)
List View
Tree View
Table View
Column View
Item Widgets (Item-Based)
List Widget
Tree Widget
Table Widget
Containers
Пишите здесь
Фильтр
Имя
Используется
Текст
Горячая клавиш
Триггерн
Редактор действий
Редактор сигналов и слотов
Объект
Класс
MainWindow
QMainWindow
centralWidget
QWidget
menuBar
QMenuBar
mainToolBar
QToolBar
statusBar
QStatusBar
Фильтр
MainWindow : QMainWindow
Свойство
Значение
QObject
objectName
MainWindow
QWidget
windowModality
NonModal
enabled
geometry
[(0, 0), 400 x 300]
sizePolicy
[Preferred, Pref..
minimumSize
0 x 0
maximumSize
16777215 x 1677.
sizeIncrement
0 x 0
baseSize
0 x 0
palette
font
[MS Shell D.
cursor
Arrow
mouseTracking
focusPolicy
NoFocus
contextMenuPolicy
DefaultContext..
Быстрый поиск (Ctrl+K)
1 Проблемы
2 Результаты поиска
3 Вывод приложения
4 Консоль сборки
5 Консоль отладчика

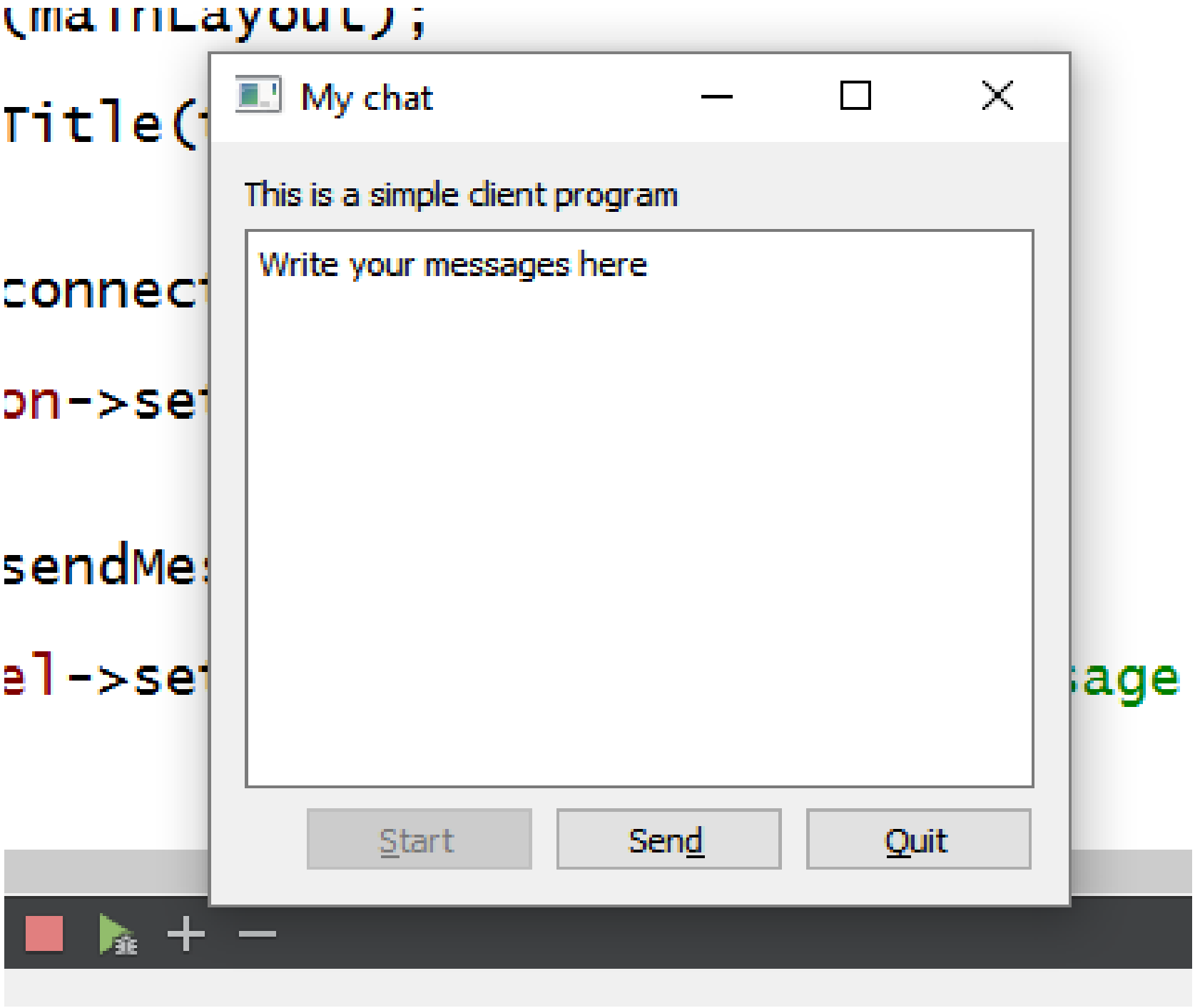

mainwindow.h - chat_client - Qt Creator
Файл Правка Сборка Отладка Анализ Инструменты Окно Справка
Проекты
chat_client
chat_client.pro
Заголовочные
mainwindow.h
Исходники
main.cpp
mainwindow.cpp
Формы
mainwindow.ui
mainwindow.h
<Выберите символ>
Строка: 1, Столбец: 1
#ifndef MAINWINDOW_H
#define MAINWINDOW_H
#include <QMainWindow>
namespace Ui {
class MainWindow;
}
class MainWindow : public QMainWindow
{
Q_OBJECT
MainWindow
Начало
Редактор
Дизайн
Отладка
Проекты
Справка
Открытые документы
main.cpp
mainwindow.cpp
mainwindow.h
mainwindow.ui
chat_client
Отладка
Быстрый поиск (Ctrl+K)
1 Проблемы
2 Результаты поиска
3 Вывод приложения
4 Консоль сборки
5 Консоль отладчика

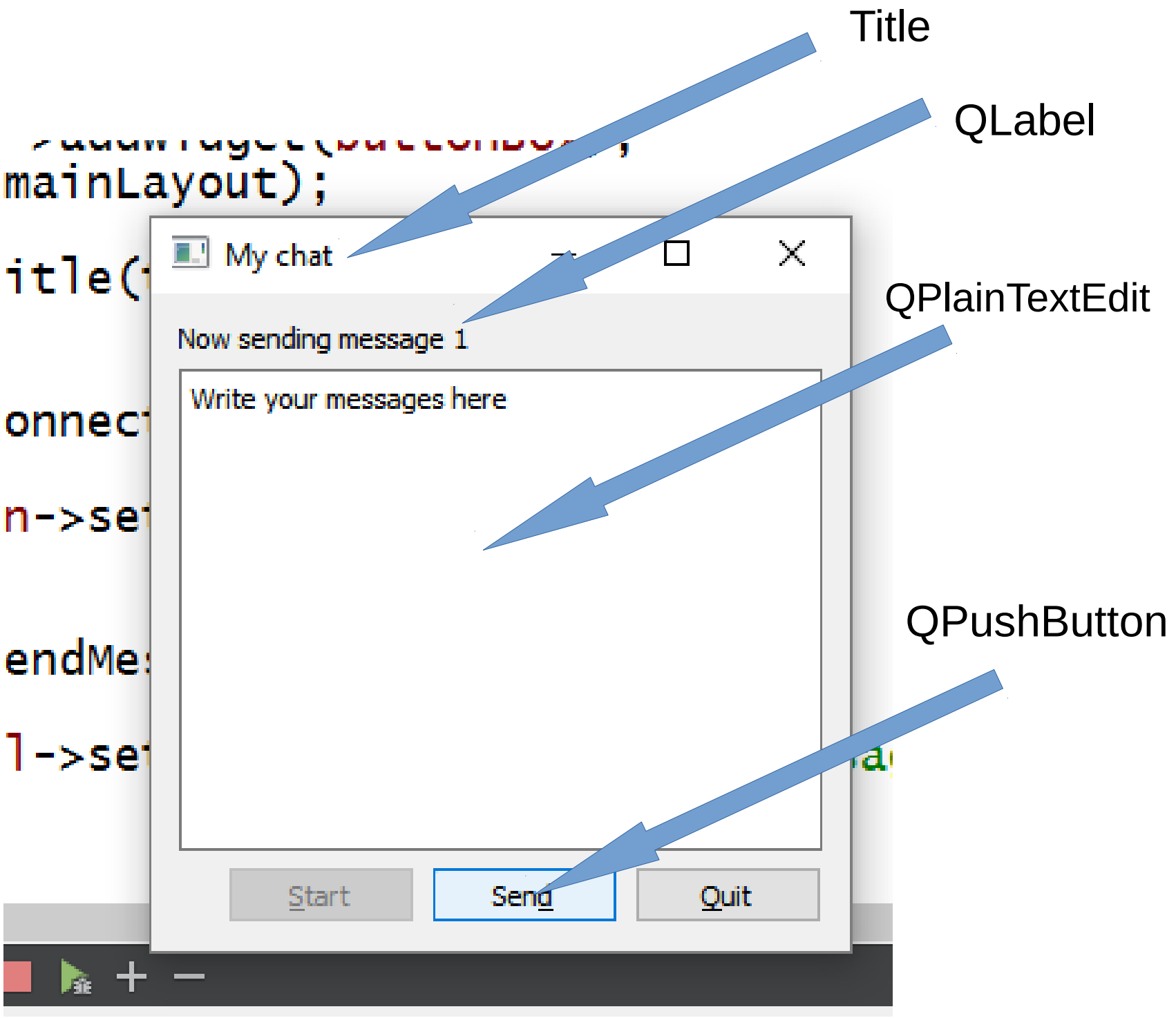
Вернемся к созданию нового проекта и выберем первым базовым классом QWidget

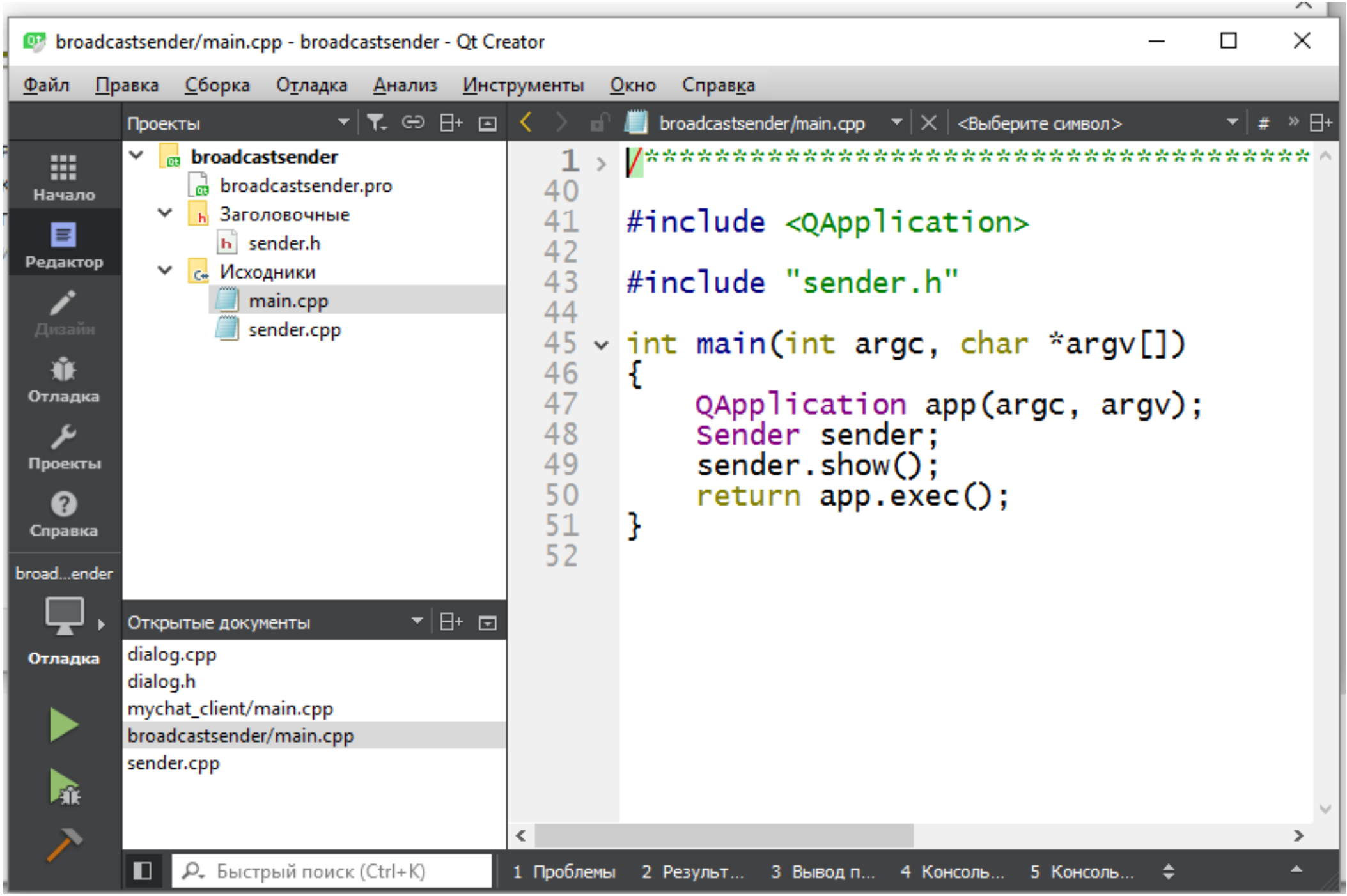

Приложение Qt Widgets
Размещение
Комплекты
Подробнее
Итог
Информация о классе
Укажите базовую информацию о классах, для которых желаете создать шаблоны файлов исходных текстов.
Имя класса:
MainWindow
Базовый класс:
QMainWindow
QMainWindow
QWidget
QDialog
Заголовочный файл:
Файл исходников:
mainwindow.cpp
Создать форму:
Файл формы:
mainwindow.ui
Далее
Отмена

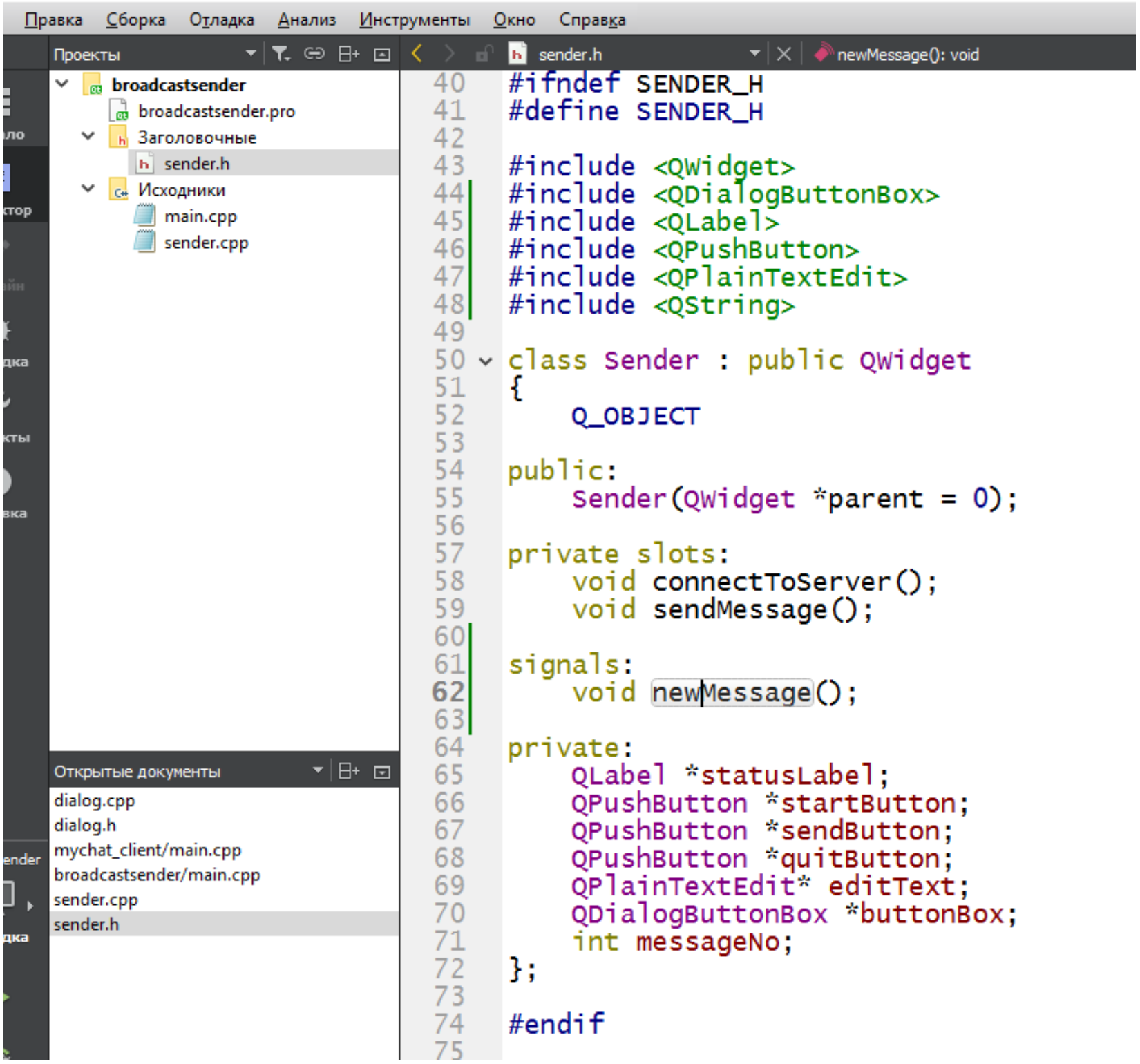
```
mainLayout->addWidget(statusLabel);
mainLayout->addWidget(editText);
mainLayout->addWidget(buttonBox);
setLayout(mainLayout);

setWindowTitle(t

oid Sender::connect

startButton->set

oid Sender::sendMe

statusLabel->set
```

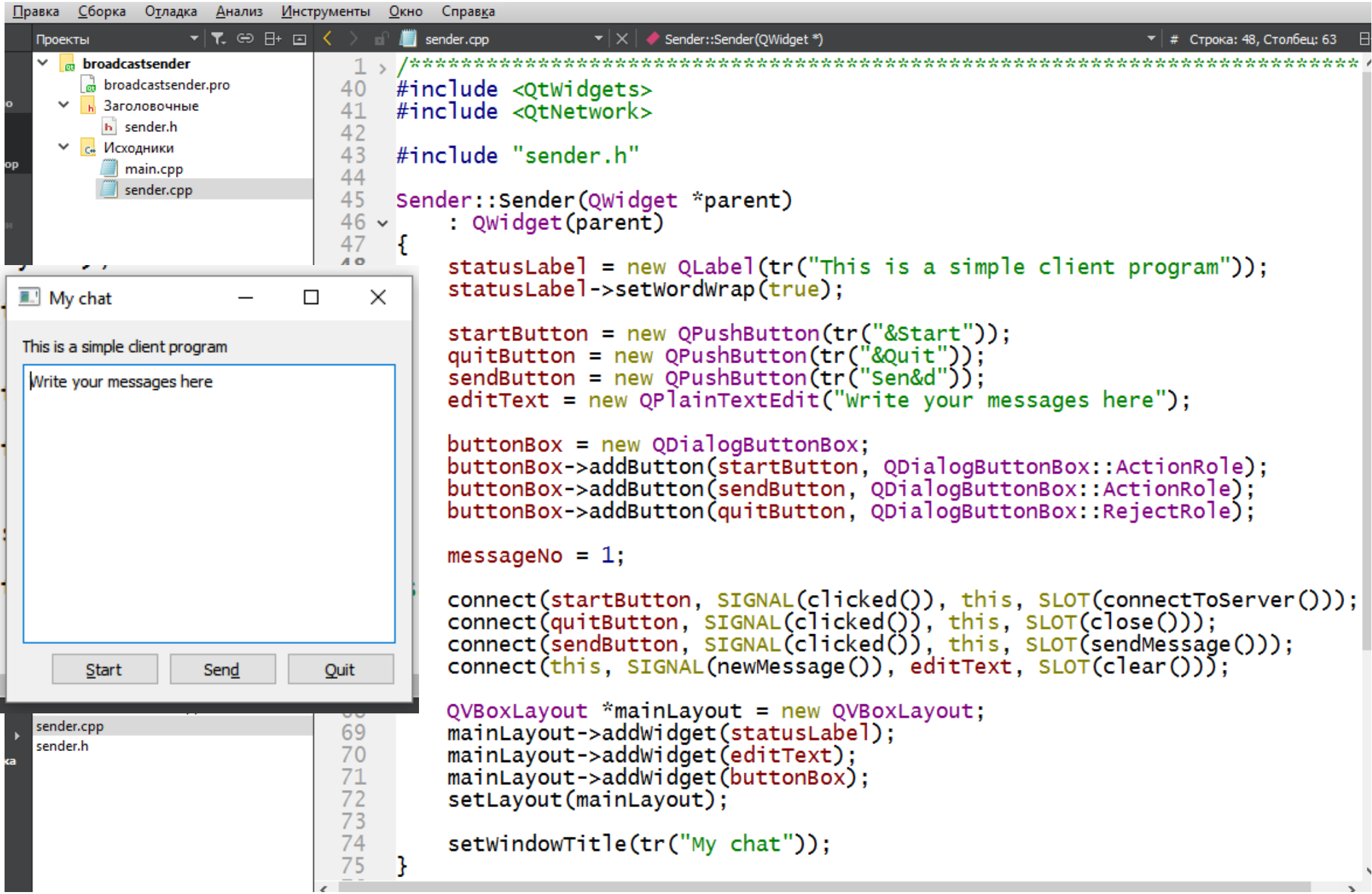


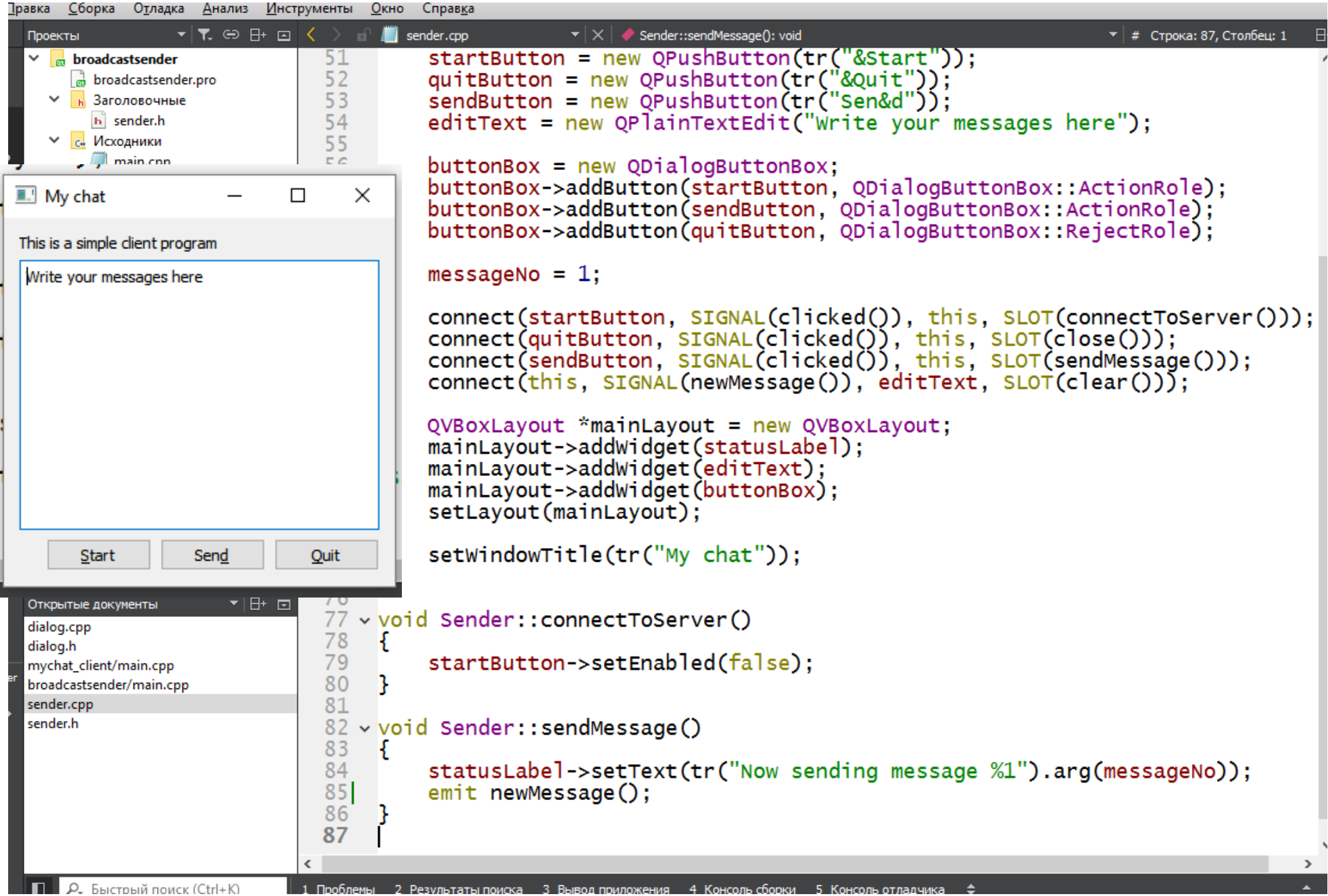

My chat
This is a simple client program
Write your messages here
Start
Send
Quit


Title
QLabel
QPlainTextEdit
QPushButton
My chat
Now sending message 1
Write your messages here
Start
Send
Quit

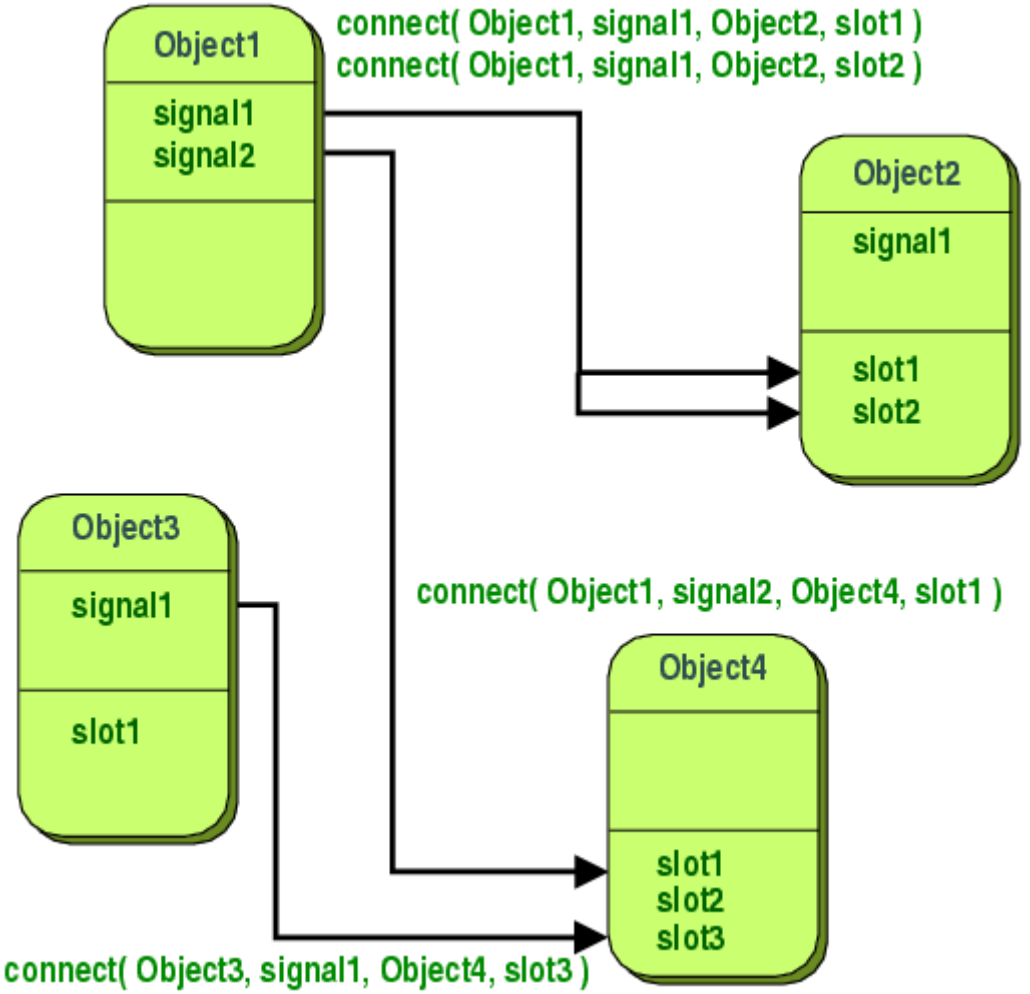

broadcastsender/main.cpp - broadcastsender - Qt Creator
Файл Правка Сборка Отладка Анализ Инструменты Окно Справка
Проекты
broadcastsender
broadcastsender.pro
Заголовочные
sender.h
Исходники
main.cpp
sender.cpp
Начало
Редактор
Дизайн
Отладка
Проекты
Справка
broad...ender
Отладка
Открытые документы
dialog.cpp
dialog.h
mychat_client/main.cpp
broadcastsender/main.cpp
sender.cpp
broadcastsender/main.cpp
<Выберите символ>
1 /**********
40
41 #include <QApplication>
42
43 #include "sender.h"
44
45 int main(int argc, char *argv[])
46 {
47 QApplication app(argc, argv);
48 Sender sender;
49 sender.show();
50 return app.exec();
51 }
52
Быстрый поиск (Ctrl+K)
1 Проблемы
2 Результ...
3 Вывод п...
4 Консоль...
5 Консоль...


Правка Сборка Отладка Анализ Инструменты Окно Справка
Проекты
broadcastsender
broadcastsender.pro
Заголовочные
sender.h
Исходники
main.cpp
sender.cpp
Открытые документы
dialog.cpp
dialog.h
mychat_client/main.cpp
broadcastsender/main.cpp
sender.cpp
sender.h
sender.h
newMessage(): void
40 #ifndef SENDER_H
41 #define SENDER_H
42
43 #include <QWidget>
44 #include <QDialogButtonBox>
45 #include <QLabel>
46 #include <QPushButton>
47 #include <QPlainTextEdit>
48 #include <QString>
49
50 class Sender : public QWidget
51 {
52 Q_OBJECT
53
54 public:
55 Sender(QWidget *parent = 0);
56
57 private slots:
58 void connectToServer();
59 void sendMessage();
60
61 signals:
62 void newMessage();
63
64 private:
65 QLabel *statusLabel;
66 QPushButton *startButton;
67 QPushButton *sendButton;
68 QPushButton *quitButton;
69 QPlainTextEdit* editText;
70 QDialogButtonBox *buttonBox;
71 int messageNo;
72 };
73
74 #endif
75


Правка Сборка Отладка Анализ Инструменты Окно Справка
Проекты
broadcastsender
broadcastsender.pro
Заголовочные
sender.h
Исходники
main.cpp
sender.cpp
sender.cpp
Sender::Sender(QWidget *)
Строка: 48, Столбец: 63
My chat
This is a simple client program
Write your messages here
Start
Send
Quit
sender.cpp
sender.h
/*****
#include <QtWidgets>
#include <QtNetwork>
#include "sender.h"
Sender::Sender(QWidget *parent)
: QWidget(parent)
{
statusLabel = new QLabel(tr("This is a simple client program"));
statusLabel->setWordWrap(true);
startButton = new QPushButton(tr("&Start"));
quitButton = new QPushButton(tr("&Quit"));
sendButton = new QPushButton(tr("Sen&d"));
editText = new QPlainTextEdit("Write your messages here");
buttonBox = new QDialogButtonBox;
buttonBox->addButton(startButton, QDialogButtonBox::ActionRole);
buttonBox->addButton(sendButton, QDialogButtonBox::ActionRole);
buttonBox->addButton(quitButton, QDialogButtonBox::RejectRole);
messageNo = 1;
connect(startButton, SIGNAL(clicked()), this, SLOT(connectToServer()));
connect(quitButton, SIGNAL(clicked()), this, SLOT(close()));
connect(sendButton, SIGNAL(clicked()), this, SLOT(sendMessage()));
connect(this, SIGNAL(newMessage()), editText, SLOT(clear()));
QVBoxLayout *mainLayout = new QVBoxLayout;
mainLayout->addWidget(statusLabel);
mainLayout->addWidget(editText);
mainLayout->addWidget(buttonBox);
setLayout(mainLayout);
setWindowTitle(tr("My chat"));
}

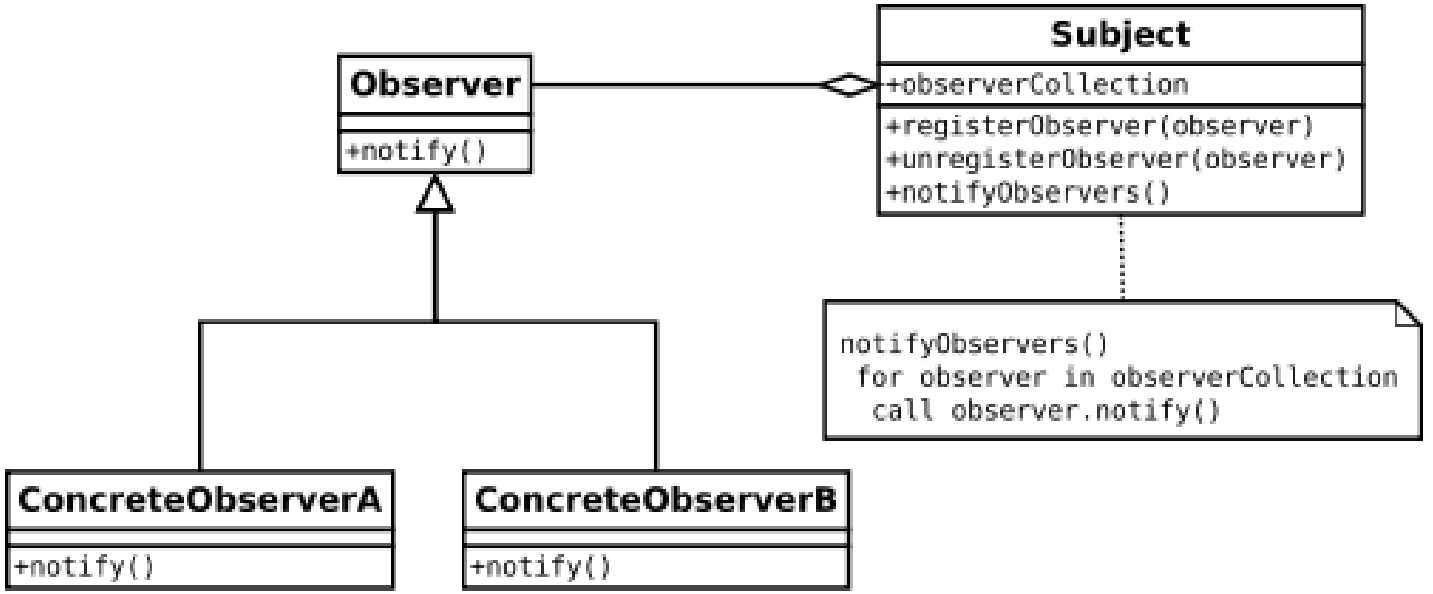

Правка Сборка Отладка Анализ Инструменты Окно Справка
Проекты
broadcastsender
broadcastsender.pro
Заголовочные
sender.h
Исходники
main.cpp
sender.cpp
Sender::sendMessage(): void
Строка: 87, Столбец: 1
My chat
This is a simple client program
Write your messages here
Start
Send
Quit
Открытые документы
dialog.cpp
dialog.h
mychat_client/main.cpp
broadcastsender/main.cpp
sender.cpp
sender.h
Быстрый поиск (Ctrl+K)
1 Проблемы 2 Результаты поиска 3 Вывод приложения 4 Консоль сборки 5 Консоль отладчика
51 startButton = new QPushButton(tr("&Start"));
52 quitButton = new QPushButton(tr("&Quit"));
53 sendButton = new QPushButton(tr("Sen&d"));
54 editText = new QPlainTextEdit("Write your messages here");
55
buttonBox = new QDialogButtonBox;
buttonBox->addButton(startButton, QDialogButtonBox::ActionRole);
buttonBox->addButton(sendButton, QDialogButtonBox::ActionRole);
buttonBox->addButton(quitButton, QDialogButtonBox::RejectRole);
messageNo = 1;
connect(startButton, SIGNAL(clicked()), this, SLOT(connectToServer()));
connect(quitButton, SIGNAL(clicked()), this, SLOT(close()));
connect(sendButton, SIGNAL(clicked()), this, SLOT(sendMessage()));
connect(this, SIGNAL(newMessage()), editText, SLOT(clear()));
QVBoxLayout *mainLayout = new QVBoxLayout;
mainLayout->addWidget(statusLabel);
mainLayout->addWidget(editText);
mainLayout->addWidget(buttonBox);
setLayout(mainLayout);
setWindowTitle(tr("My chat"));
77 void Sender::connectToServer()
78 {
79 startButton->setEnabled(false);
80 }
81
82 void Sender::sendMessage()
83 {
84 statusLabel->setText(tr("Now sending message %1").arg(messageNo));
85 emit newMessage();
86 }
87

Сигналы и слоты

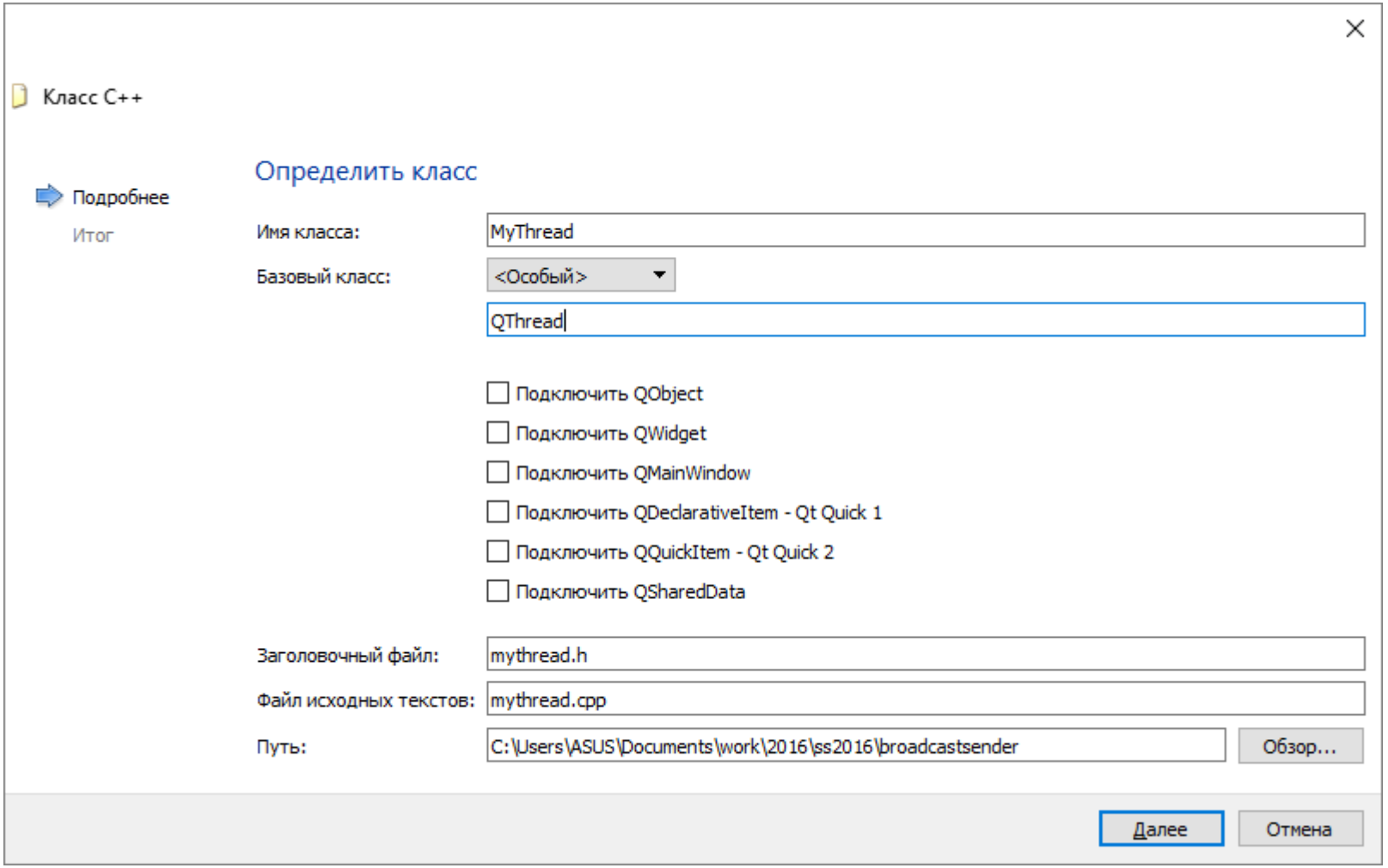


Изображение: http://doc.qt.io/qt-4.8/signalsandslots.html

Сигналы и слоты: наблюдатель (шаблон проектирования)



Изображение:
https://en.wikipedia.org/wiki/Observer_pattern


Класс C++
Подробнее
Итог
Определить класс
Имя класса:
MyThread
Базовый класс:
<Особый>
QThread
Подключить QObject
Подключить QWidget
Подключить QMainWindow
Подключить QDeclarativeItem - Qt Quick 1
Подключить QQuickItem - Qt Quick 2
Подключить QSharedData
Заголовочный файл:
mythread.h
Файл исходных текстов:
mythread.cpp
Путь:
C:\Users\ASUS\Documents\work\2016\ss2016\broadcastsender
Обзор...
Далее
Отмена

События: events

```
#ifndef KEYEVENT_H
#define KEYEVENT_H

#include <QWidget>
#include <QtGui>

class KeyEvent : public QWidget
{
    Q_OBJECT
public:
    KeyEvent(QWidget *parent = 0);

protected:
    void keyPressEvent(QKeyEvent *);

private:
    QLabel *label;
    QVBoxLayout *layout;
};

#endif // KEYEVENT_H
```

```
#include "keyevent.h"

#include <QApplication>
#include <QKeyEvent>

KeyEvent::KeyEvent(QWidget *parent) :
    QWidget(parent)
{
    myLabel = new QLabel("Press a key");
    mainLayout = new QVBoxLayout;
    mainLayout->addWidget(label);
    setLayout(layout);

}

void KeyEvent::keyPressEvent(QKeyEvent *event)
{
       label->setText("You pressed"+event->text());
}
```

```
#include <QtGui>
#include "keyevent.h"

int main(int argc, char *argv[])
{
    QApplication a(argc, argv);

    KeyEvent *keyEvent = new KeyEvent();
    keyEvent->show();

    return a.exec();
}
```

Groups Of Related Classes

This is a list of functional groups of Qt classes. A class can appear in more than one functional group.

Container Classes	Qt's template-based container classes.
Event Classes	Classes used to create and handle events
Help System	Classes used to provide online-help for applications
Input/Output and Networking	Classes providing file input and output along with directory and network handling
Network Programming API	Classes for Network Programming
Painting Classes	Classes that provide support for painting
Plugin Classes	Plugin related classes
Printer and Printing APIs	Classes for producing printed output
Rendering in 3D	Classes that provide support for rendering in 3D