

# Qt Designer

Дмитрий Фёдоров  
Руководитель отдела разработки ПО



# Проверка связи



## Если у вас нет звука:

- убедитесь, что на вашем устройстве и на колонках включён звук
- обновите страницу вебинара (или закройте страницу и заново присоединитесь к вебинару)
- откройте вебинар в другом браузере
- перезагрузите компьютер (ноутбук) и заново попытайтесь зайти



## Поставьте в чат:

-  если меня видно и слышно
-  если нет

# Дмитрий Фёдоров

О спикере:

- более 10 лет в разработке авиационных систем
- возглавляет отдел разработки ПО в НИЦ «ИРТ»



# Вспоминаем прошрое занятие

**Вопрос:** что такое модуль Qt?



# Вспоминаем прошное занятие

**Вопрос:** что такое модуль Qt?

**Ответ:** это набор классов



# Вспоминаем прошное занятие

**Вопрос:** какой модуль Qt является базовым?



# Вспоминаем прошрое занятие

**Вопрос:** какой модуль Qt является базовым?

**Ответ:** QtCore



# Вспоминаем прошрое занятие

**Вопрос:** какой модуль Qt отвечает за работу с сетевыми запросами?





# Вспоминаем прошрое занятие

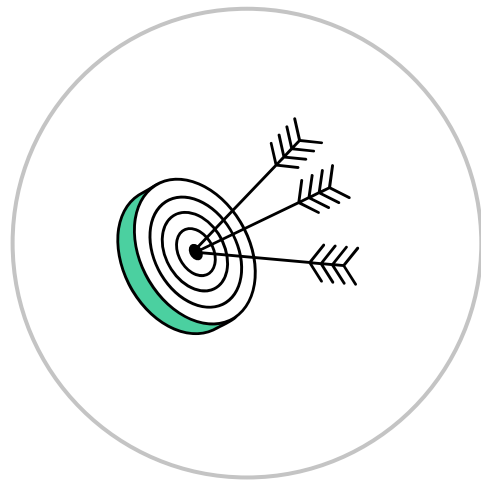
**Вопрос:** какой модуль Qt отвечает за работу с сетевыми запросами?

**Ответ:** QtNetwork



# Цели занятия

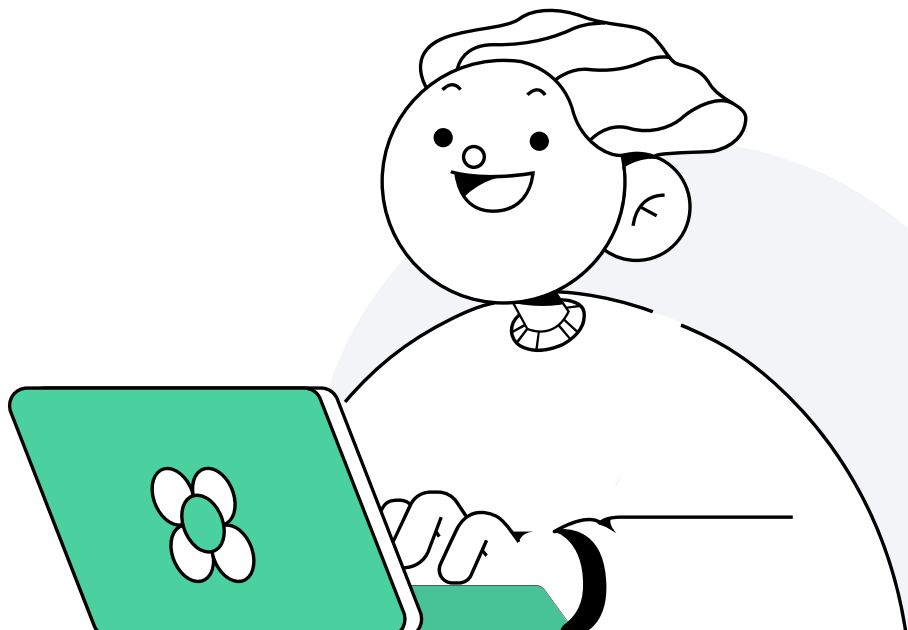
- Узнаем, что такое Qt Designer
- Научимся создавать простые графические формы



# План занятия

- 1 Что такое Qt Designer
- 2 Основная рабочая область Qt Designer
- 3 Редактирование окна
- 4 Итоги
- 5 Домашнее задание

\*Нажми на нужный раздел для перехода



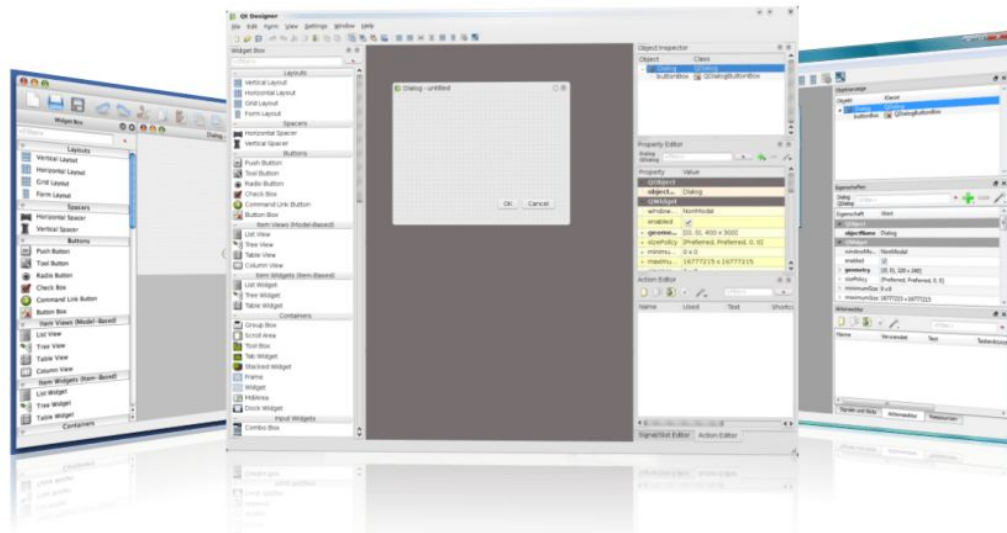
# Что такое Qt Designer



1

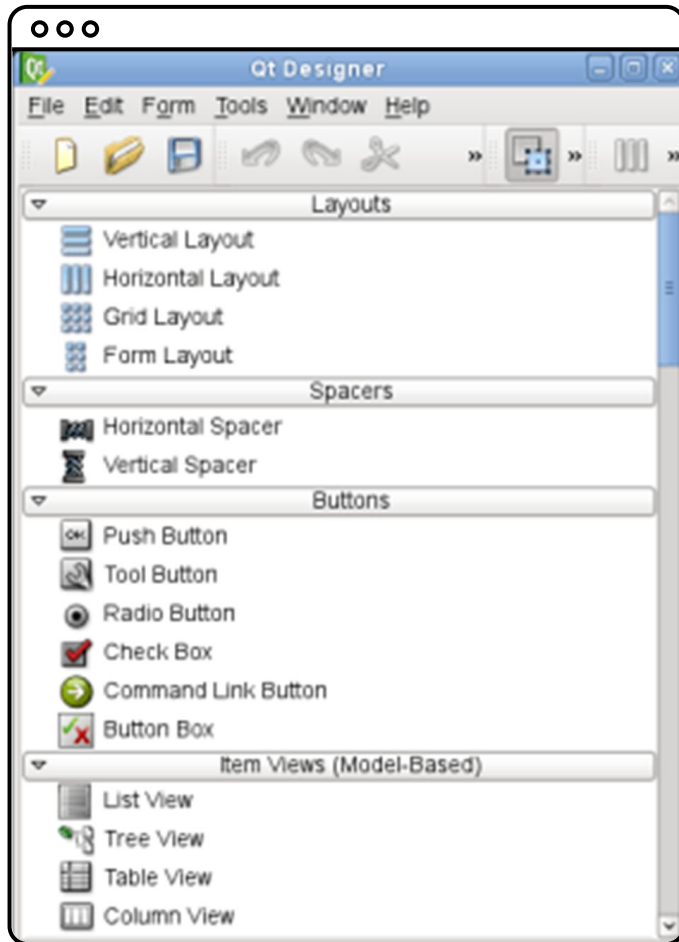
# Qt Designer

→ **Qt Designer** — инструмент для проектирования и создания графических пользовательских интерфейсов (GUI) из компонентов Qt



# Qt Designer

- **Автономный режим работы** — подходит для UI-дизайнеров, создающих пользовательский интерфейс приложений
- **В составе Qt Creator** — запускается из проекта, редактируемые формы уже включены в проект



# Основная рабочая область Qt Designer



2

# Форма Qt

В приложении Qt Widgets присутствует файл формы с расширением \*.ui, представляющий собой дерево виджетов формы в формате XML.

## Формы могут быть обработаны:

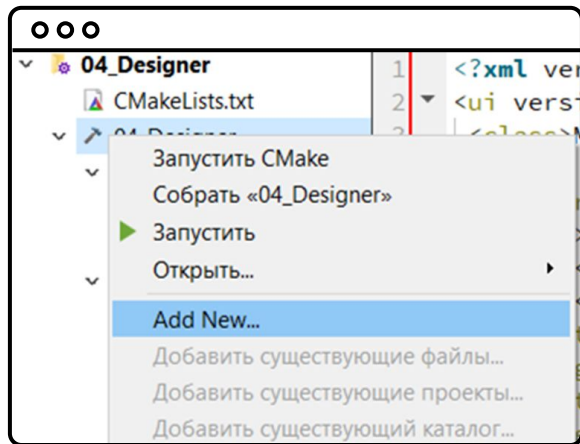
- во время компиляции — это означает, что формы будут преобразованы в код C++, который возможно скомпилировать
- во время выполнения — это означает, что формы обрабатываются классом QUiLoader, который динамически создаёт дерево виджетов при анализе XML-файла

```
1 <?xml version="1.0" encoding="UTF-8"?>
2 <ui version="4.0">
3   <class>MainWindow</class>
4   <widget class="QMainWindow" name="MainWindow">
5     <property name="geometry">
6       <rect>
7         <x>0</x>
8         <y>0</y>
9         <width>667</width>
10        <height>527</height>
11      </rect>
12    </property>
13    <property name="windowTitle">
14      <string>MainWindow</string>
15    </property>
16    <widget class="QWidget" name="centralwidget"/>
17    <widget class="QMenuBar" name="menubar">
18      <property name="geometry">
19        <rect>
20          <x>0</x>
```

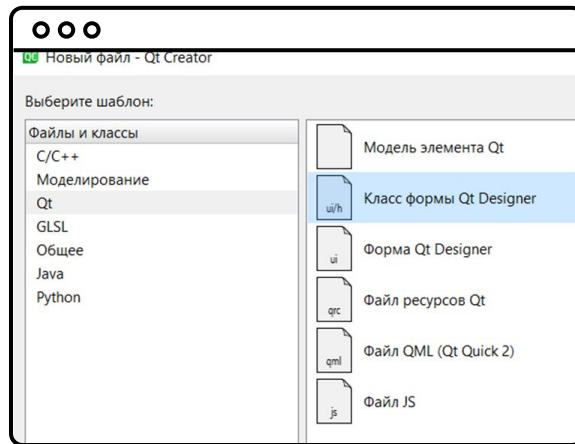


# Создание новой формы

1 Щёлкните правой кнопкой мыши по проекту



2 Выберите класс формы Qt Designer



При компиляции проекта форма преобразуется в экземпляр класса

# Отображение новой формы

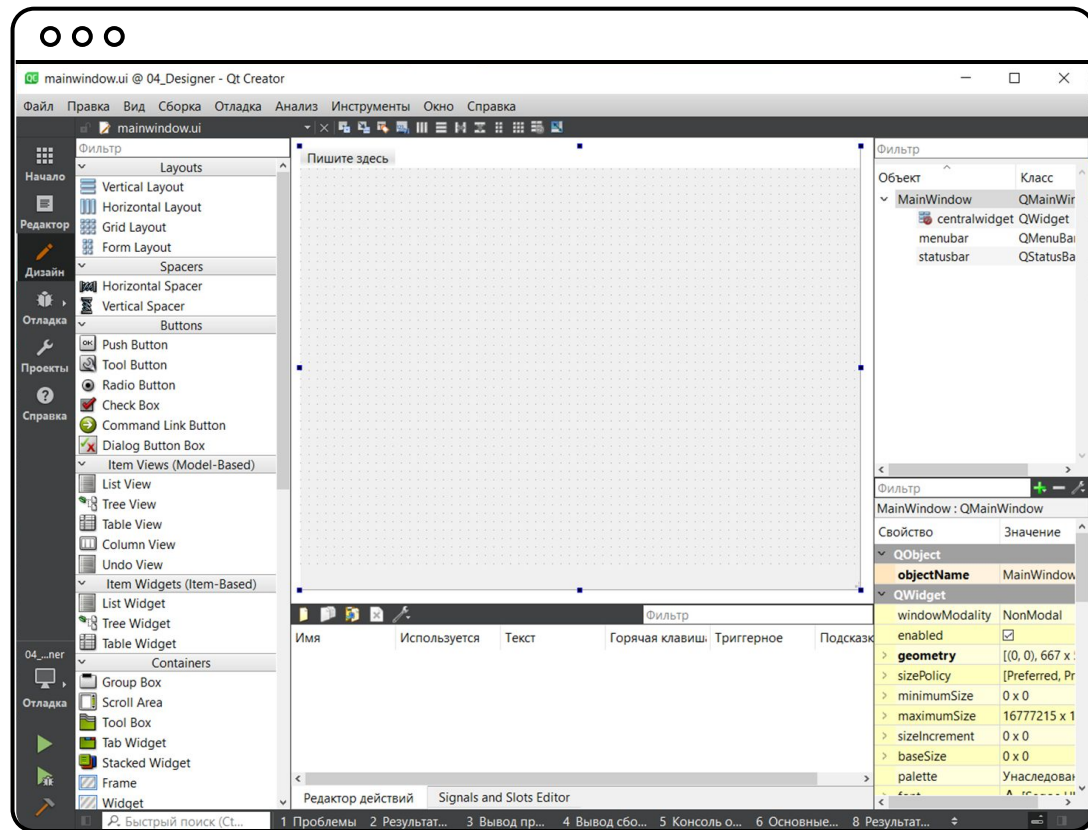
```
04_Designer
├── CMakeLists.txt
├── 04_Designer
│   ├── Header Files
│   │   ├── dialog.h
│   │   └── mainWindow.h
│   ├── Source Files
│   │   ├── dialog.cpp
│   │   ├── main.cpp
│   │   └── mainWindow.cpp
│   ├── dialog.ui
│   └── mainWindow.ui
└── CMake Modules

1  #ifndef MAINWINDOW_H
2  #define MAINWINDOW_H
3
4  #include <QMainWindow>
5  #include "dialog.h"
6  QT_BEGIN_NAMESPACE
7  namespace Ui { class MainWindow; }
8  QT_END_NAMESPACE
9
10 class MainWindow : public QMainWindow
11 {
12     Q_OBJECT
13
14 public:
15     MainWindow(QWidget *parent = nullptr);
16     ~MainWindow();
17
18 private:
19     Ui::MainWindow *ui;
20     Dialog *d; //Объявляем указатель
21               //на новую форму
22
23 };
24 #endif // MAINWINDOW_H
```

```
04_Designer
├── CMakeLists.txt
├── 04_Designer
│   ├── Header Files
│   │   ├── dialog.h
│   │   └── mainWindow.h
│   ├── Source Files
│   │   ├── dialog.cpp
│   │   ├── main.cpp
│   │   └── mainWindow.cpp
│   ├── dialog.ui
│   └── mainWindow.ui
└── CMake Modules

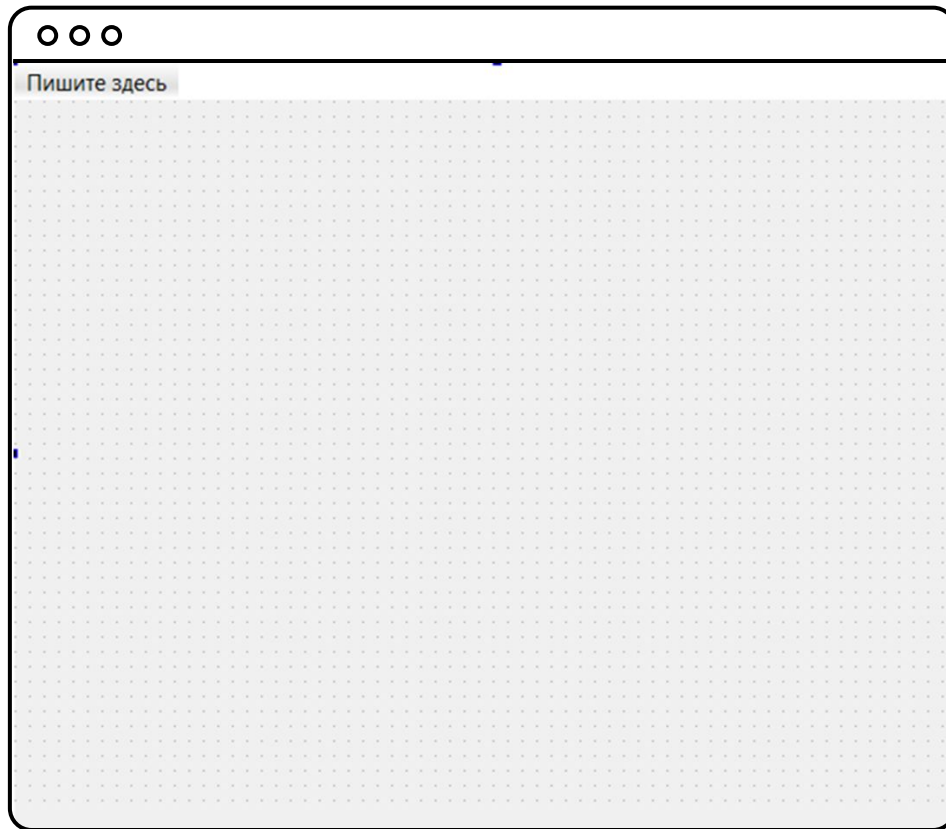
1  #include "mainwindow.h"
2  #include "./ui_mainwindow.h"
3
4  MainWindow::MainWindow(QWidget *parent)
5      : QMainWindow(parent)
6      , ui(new Ui::MainWindow)
7  {
8      ui->setupUi(this);
9
10     d = new Dialog(this); //Выделяем память
11     d->show();           //Отображаем
12 }
13
14
15 MainWindow::~MainWindow()
16 {
17     delete ui;
18 }
19
20
```

# Qt Designer. Редактор форм



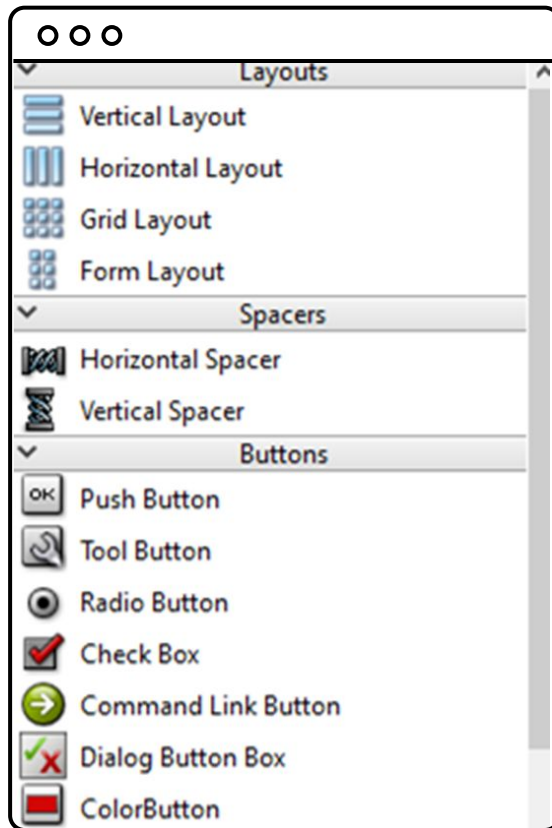
# Редактор форм

Отображается текущий вид  
пользовательской формы



# Окно виджетов

Окно виджетов предоставляет выбор стандартных виджетов Qt, компоновок и других объектов, которые могут применяться для создания пользовательских интерфейсов на формах

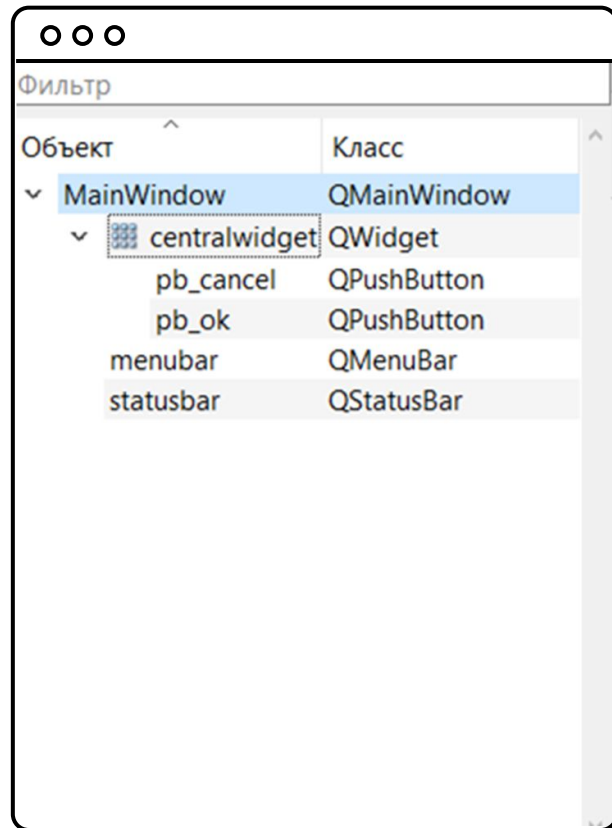


# Инспектор объектов

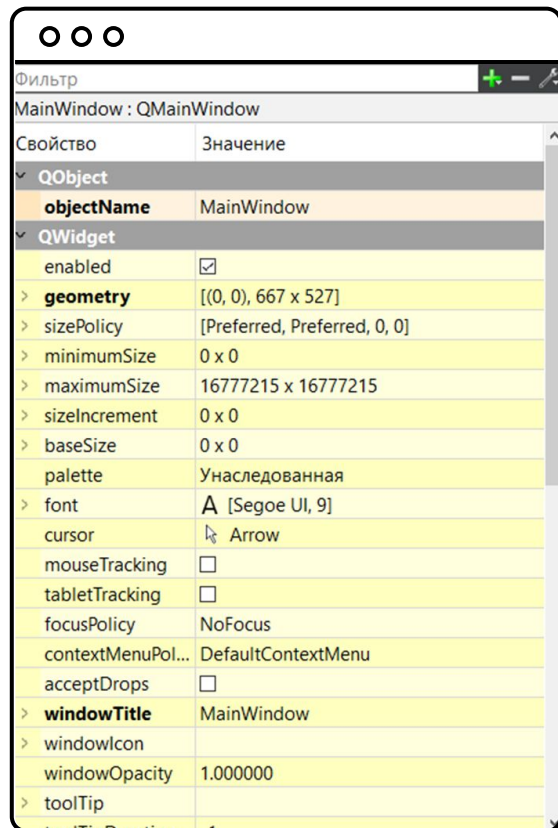
Все виджеты, присутствующие на форме, отображаются в иерархическом порядке.



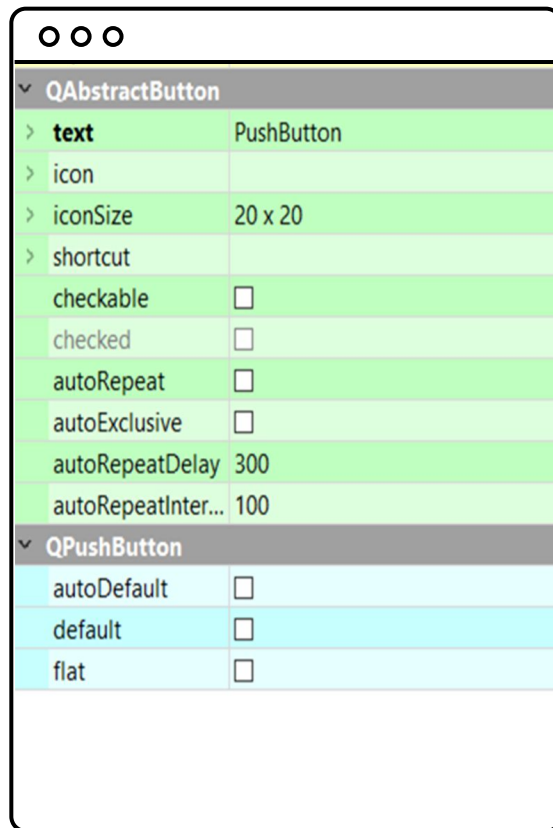
**Важно:** необходимо однотипно и понятно именовать все виджеты на форме, соблюдая те же рекомендации, что и при именовании переменных



# Окно свойств

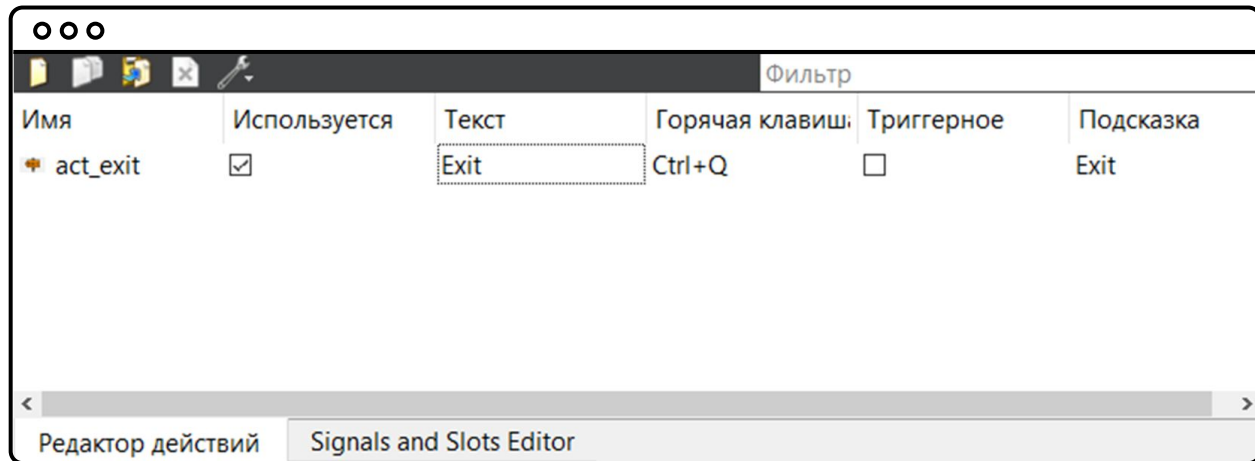
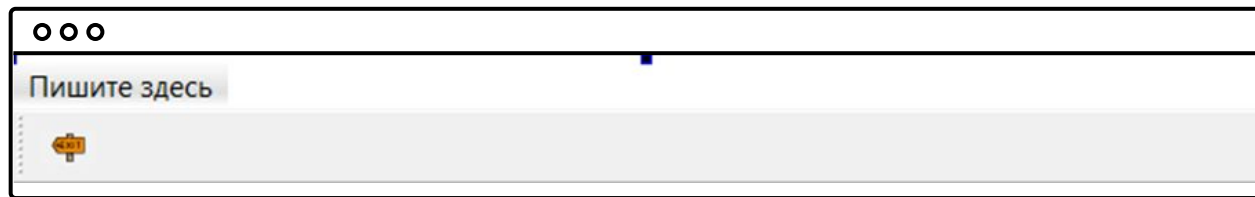


Доступны все свойства виджетов. Поскольку эти виджеты — наследники QWidget, им доступны все свойства в иерархии наследования



# Редактор действий

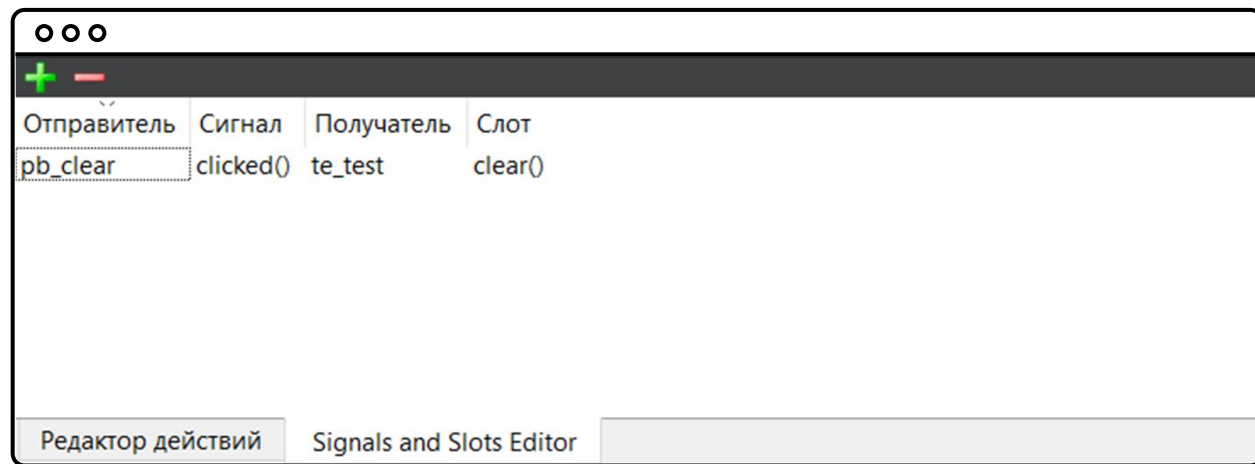
Позволяет добавлять элементарные действия на панель инструментов в приложении. Каждое действие описывается своим слотом





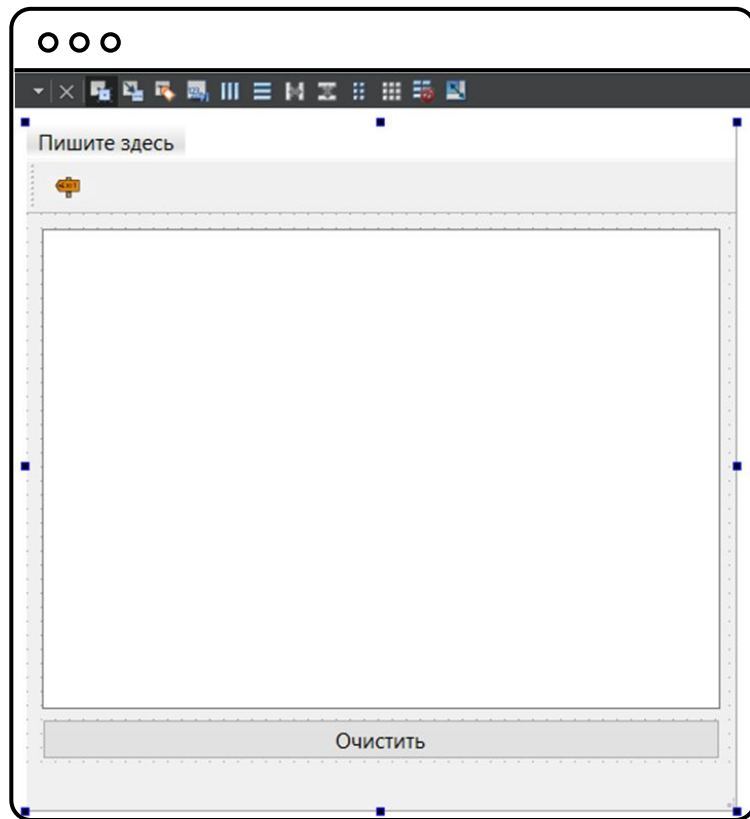
# Редактор сигналов и слотов

Даёт возможность редактировать связи между стандартными сигналами/слотами виджетов формы



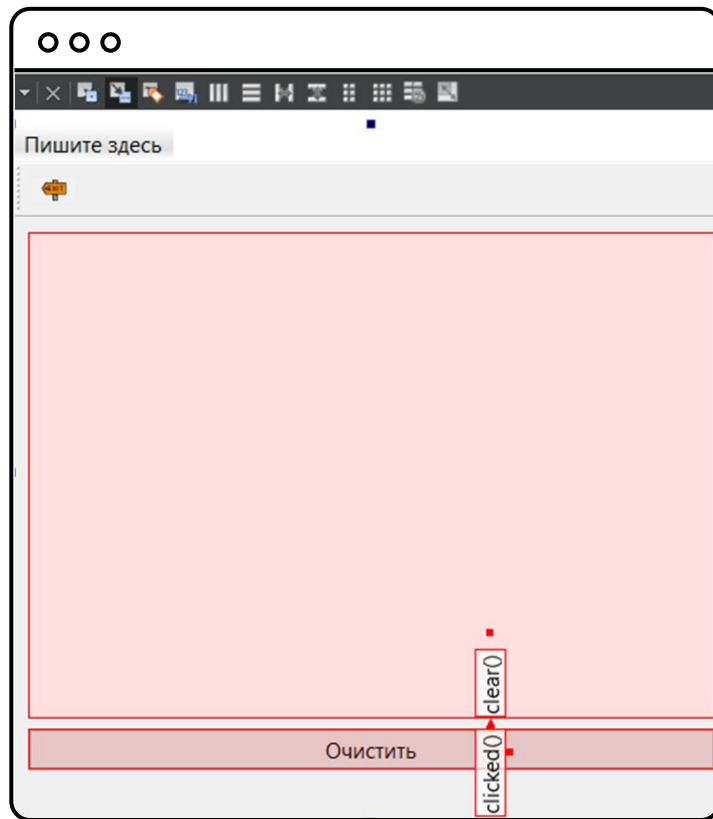
# Режим редактирования виджетов

Режим установлен по умолчанию,  
позволяет редактировать  
виджеты на форме



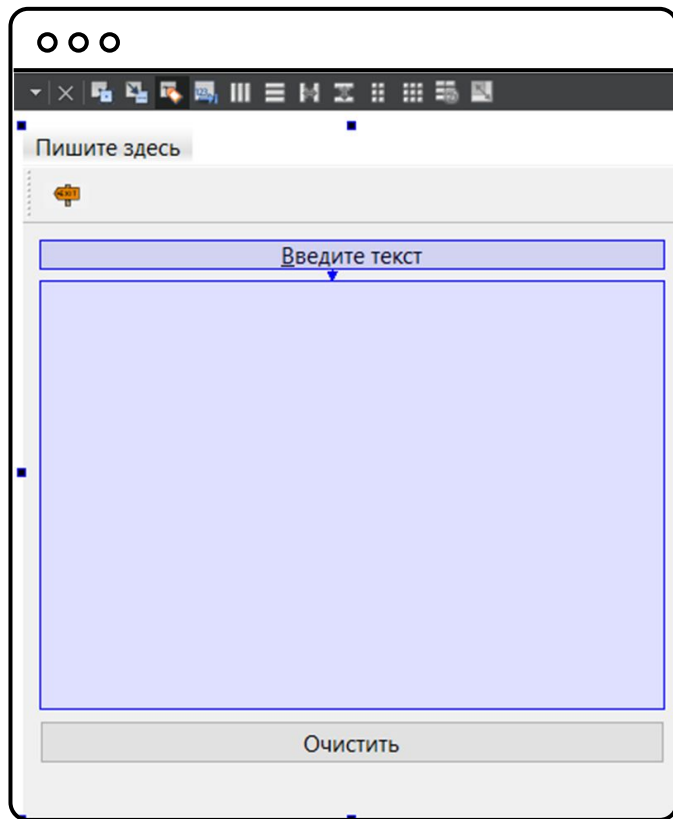
# Режим редактирования сигналов/слотов

Позволяет добавлять стандартные сигналы/слоты для обмена сообщениями между виджетами без создания дополнительного кода в редакторе



# Режим редактирования партнёров

Позволяет привязать редактируемое поле к лейблу. В случае привязки фокус переводится на редактируемое поле по нажатию горячих клавиш



# Режим изменения табулятора

Изменяет порядок фокусировки полей при нажатии клавиши Tab

ooo

Пишите здесь

✖

1 Введите текст

2 Введите число

3 Очистить

# Редактирование окна



3

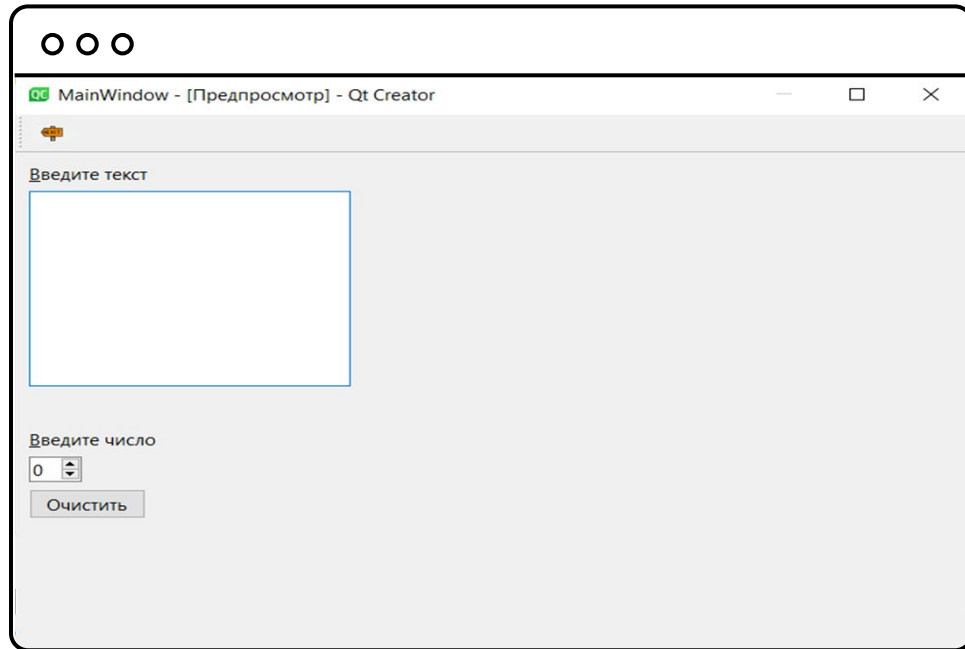
# Компоновка виджетов



Компоновка виджетов позволяет пропорционально изменять размеры виджетов и расстояние между ними при увеличении или уменьшении размеров окна. Это делает интерфейс более приятным для восприятия.

## Компоновка относится:

- к форме целиком
- к контейнерам, присутствующим в форме



# Вертикальная компоновка



Все виджеты располагаются  
вертикально

A vertical form layout within a window frame. The window has a title bar with three small circles on the left. The form is composed of several stacked components: a text input field with the placeholder text "Пишите здесь"; a small orange icon; a large text area with the placeholder text "Введите текст"; a number input field with the placeholder text "Введите число" and a value of "0"; and a "Очистить" (Clear) button at the bottom. The components are separated by thin horizontal lines and are all aligned to the left.



# Вертикальная компоновка в два ряда



Все виджеты располагаются вертикально по два элемента в ряду. Это удобно для конструкций «Название — поле»

О О О

Пишите здесь

Назад

Введите текст

Введите число 0

Очистить

# Горизонтальная компоновка



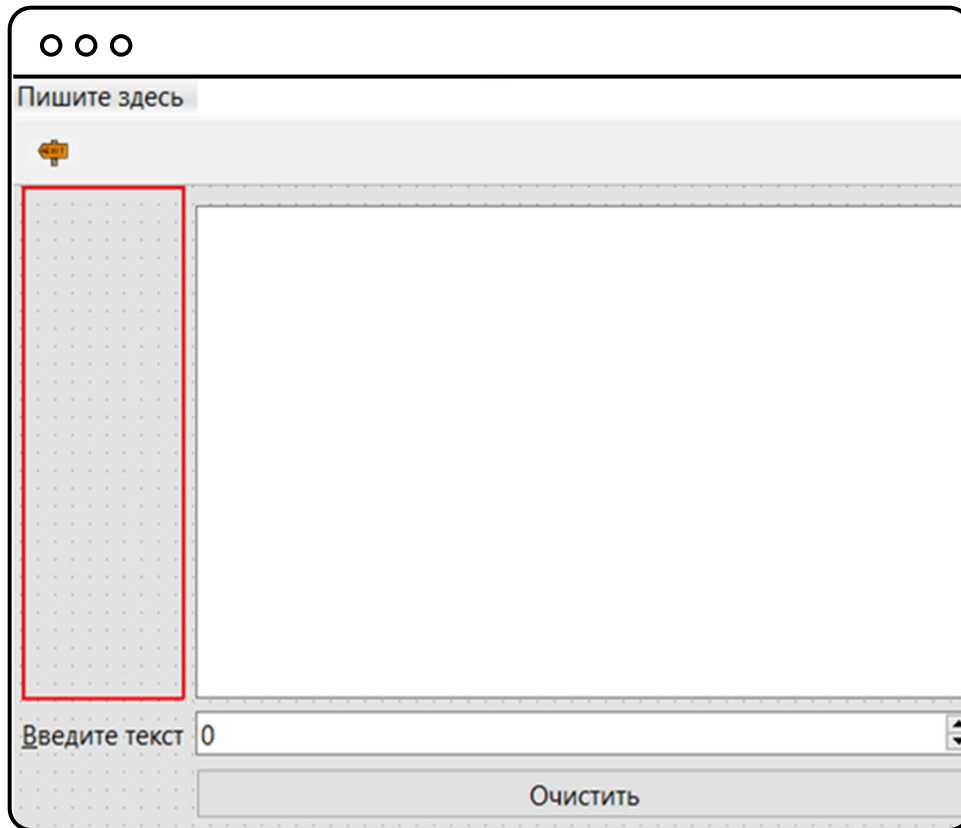
Все виджеты располагаются горизонтально

A screenshot of a web application interface. At the top, there are three small circles. Below them is a text input field with the placeholder text "Пишите здесь". Underneath the input field is a small orange icon. The main area of the interface is a large, empty rectangular box with a light gray background and a fine grid pattern. At the bottom left, there is a label "Введите число" followed by a text input field containing the number "0" and a small vertical spinner icon. To the right of this is a label "Введите текст". At the bottom right, there is a button labeled "Очистить".

# Компоновка сеткой

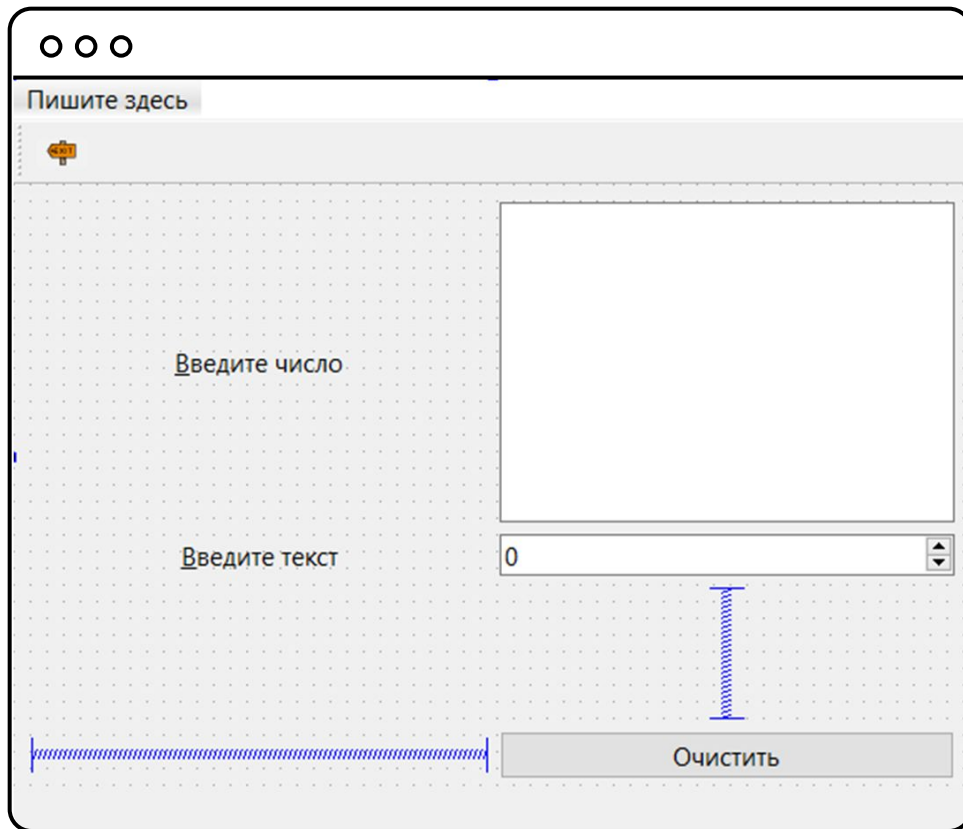


Все виджеты располагаются в узлах сетки. Каждый виджет может занимать несколько узлов



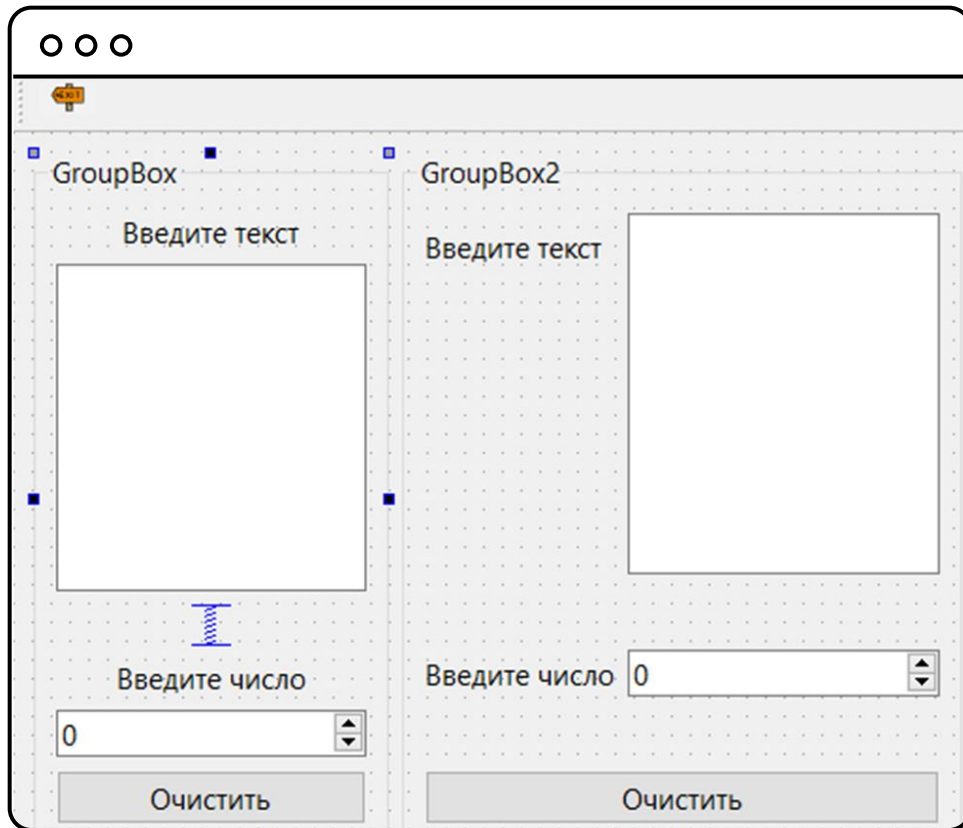
# Спейсеры (заполнители)

Спейсеры позволяют сгруппировать виджеты в определённой части формы или контейнера



# Компоновка контейнеров

В каждом контейнере используется независимая компоновка. Сами контейнеры также могут компоноваться в рамках формы



# Компоновка компоновок

Несколько виджетов могут объединяться одной компоновкой, а компоновки, в свою очередь, также могут быть объединены общей компоновкой



# Итоги

4

# Итоги занятия

- 1 Узнали, что такое Qt Designer
- 2 Рассмотрели основную рабочую область Qt Designer
- 3 Изучили редактирование окон и компоновки форм





# Домашнее задание

Давайте посмотрим ваше [домашнее задание](#).

- 1 Вопросы по домашней работе задавайте в чате группы
- 2 Задачи можно сдавать по частям
- 3 Зачёт по домашней работе ставят после того, как приняты все задачи



# Дополнительные материалы

- Описание [КОМПОНОВОК](#)
- Мануал по [Qt Designer](#)



# Qt Designer

Дмитрий Фёдоров  
Руководитель отдела разработки ПО

