

Лабораторная работа №9. Использование библиотеки Qt

Введение

Работа посвящена изучению возможностей библиотеки Qt. Вся необходимая информация в соответствующей лекции.

Создание приложения с графическим интерфейсом на Qt

1. Запустите QtCreator и создайте новый проект из шаблона Qt Widget, тип приложения GUI приложение Qt. В настройках проекта укажите его название и расположение, остальные настройки пока не меняем.

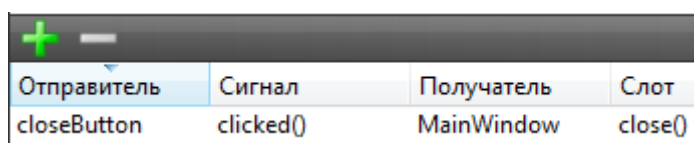
В имени файла и пути к нему не должно быть русских букв и других символов Unicode! В противном случае проект не сможет быть собран.

2. Если все сделано правильно, ваш проект будет содержать 5 файлов:
 - В `mainwindow.h` содержится описание созданного класса.
 - Файл `mainwindow.cpp` содержит этот класс.
 - Файл `main.cpp` содержит функцию `main()`, его в этом проекте трогать не будем.
 - Еще есть файл формы `mainwindow.ui`, с него и начнем.


На самом деле есть еще файл с расширением `*.pro`, который содержит описание проекта, но он пока нам не нужен.

3. Файл формы можно просматривать в двух режимах – как XML-файл и в графическом дизайнере, а редактировать только во втором. Дважды щелкните по файлу `mainwindow.ui` и перейдите к форме.
4. Слева находится панель виджетов (визуальных элементов). Поместите на форму одну кнопку Push Button, измените ее название на `closeButton`, а текст на `Close`. Теперь необходимо сделать так, чтобы она закрывала приложение. Для этого используем редактор сигналов и слотов

(внизу экрана под формой нужно выбрать соответствующую закладку). Нажмите **+**. В появившейся строке необходимо выбрать следующие настройки:

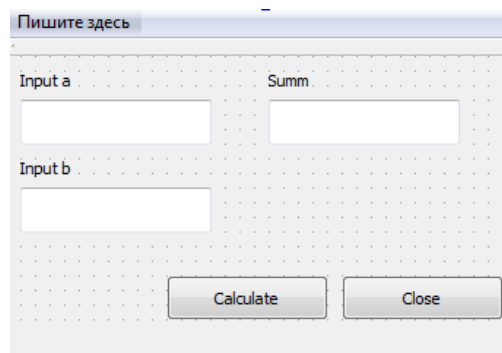


Отправитель	Сигнал	Получатель	Слот
closeButton	clicked()	MainWindow	close()

5. Теперь попробуйте запустить приложение с помощью  (слева внизу). Кнопка должна работать.
6. Добавьте на форму еще один элемент Push Button, три элемента Line Edit и три элемента Label. Приведите форму к следующему виду, не забыв переименовать объекты с помощью инспектора объектов в правой части экрана:

Объект	Класс
MainWindow	QMainWindow
centralWidget	QWidget
closeButton	QPushButton
lEditSum	QLineEdit
lEdita	QLineEdit
lEditb	QLineEdit
label	QLabel
label_2	QLabel
label_3	QLabel
summButton	QPushButton
menuBar	QMenuBar
mainToolBar	QToolBar
statusBar	QStatusBar

Форма должна выглядеть примерно так:



7. Осталось написать код, который при нажатии на кнопку Calculate (имя SummButton) вычисляет сумму из полей ввода lEdita и lEditb и записывает ее в lEditSum. В терминологии слотов и сигналов Qt это означает, что нам необходимо обработать сигнал нажатия кнопки и запустить слот, решающий задачу.
8. Добавим в класс (в заголовочный файл) два метода:
 - Публичный метод `void onClickedSummButton();`, который свяжет сигнал и слот.
 - Публичный слот `void getsum();`, который и будет вычислять сумму.
9. Добавим в файл с телом класса соответствующие методы:

```

18 void MainWindow::getsum()
19 {
20     int a,b;
21     a=ui->lEdita->text().toInt();
22     b=ui->lEditb->text().toInt();
23     QString qqq=QString::number(a+b);
24     ui->lEditSum->setText(qqq);
25 }

```

Можно обойтись без дополнительных переменных, но так код станет менее понятным.

- Метод, который свяжет встроенный сигнал нажатия на кнопку и написанную нами функцию:

```

27 void MainWindow::onClickedSummButton() {
28     connect(ui->SummButton, SIGNAL(clicked()), this, SLOT(getsum()));
29 }

```

10. Если сейчас запустить программу, она не будет работать. Это связано с тем, что мы не вызвали метод нажатия на кнопку. Сделать это можно, например, в конструкторе, который примет следующий вид (нужно доработать существующий конструктор, а не писать новый!):

```
4 MainWindow::MainWindow(QWidget *parent) :  
5     QMainWindow(parent),  
6     ui(new Ui::MainWindow)  
7 {  
8     ui->setupUi(this);  
9     onClickedSummButton();  
10 }
```

11. Запустите программу и проверьте ее работу на разных примерах. Она работает только с целыми числами. Измените программу таким образом, чтобы она могла вычислять сумму любых действительных чисел.

Задание

Разработайте приложение-калькулятор, позволяющее выполнять различные арифметические действия.

Критерии выставления баллов:

- Базовый функционал (арифметические действия над вещественными числами) – 2 балла.
- Удобный пользовательский интерфейс (кнопки с цифрами, удобное расположение элементов, единственное поле для ввода, защита от неправильного ввода данных, обработка исключений) – 3 балла.