# Лабораторная работа 1,2

# Задание

**Создать кнопки:**

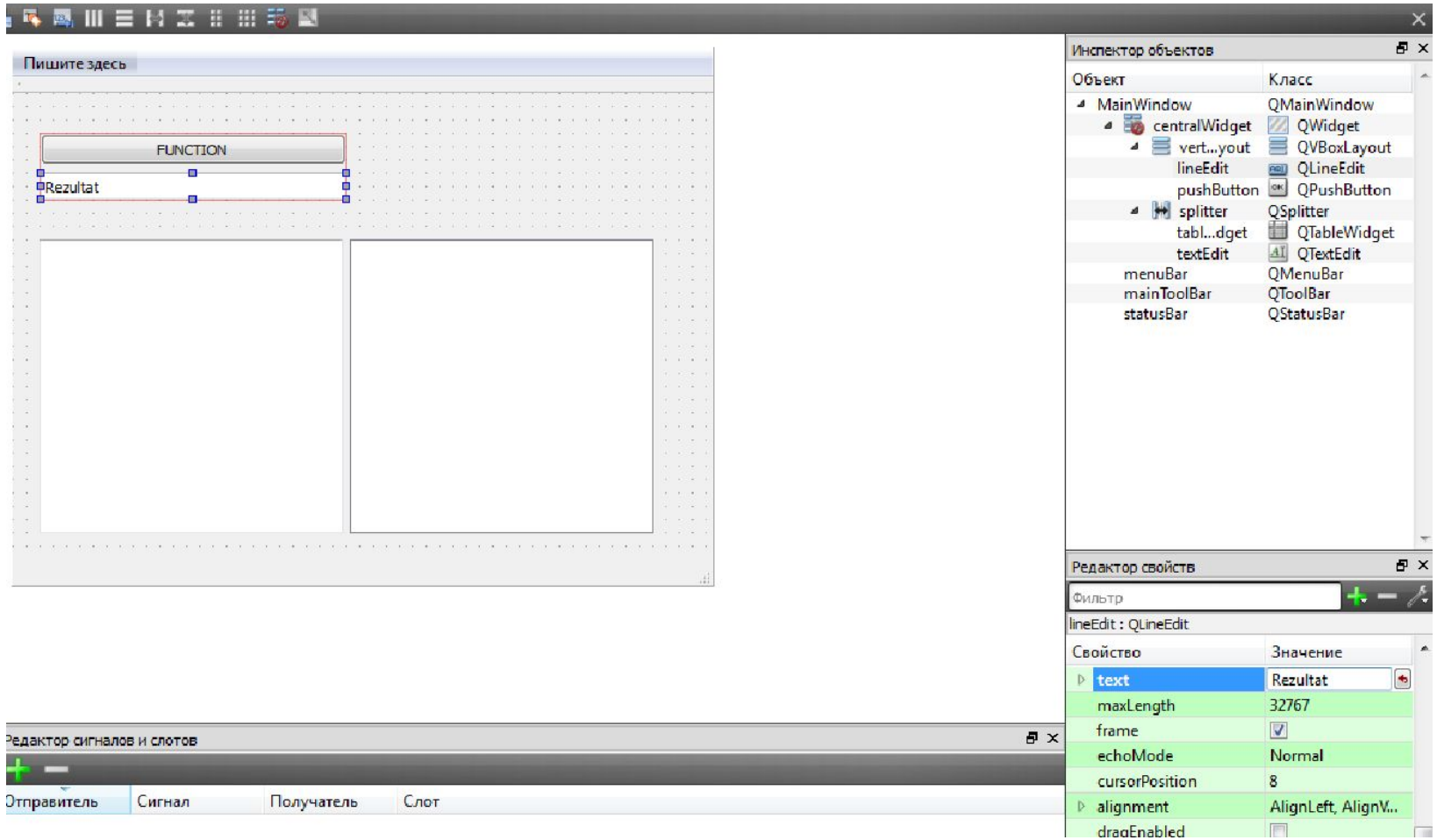
- Вывода результата функции в элемент TextLabel;
- Вывода сформированной матрицы в TextEdit;
- Вывод матрицы в таблицу;
- Очистки содержимого полей;
- Закрытия формы (режим дизайна – Редактор сигналов и слотов).

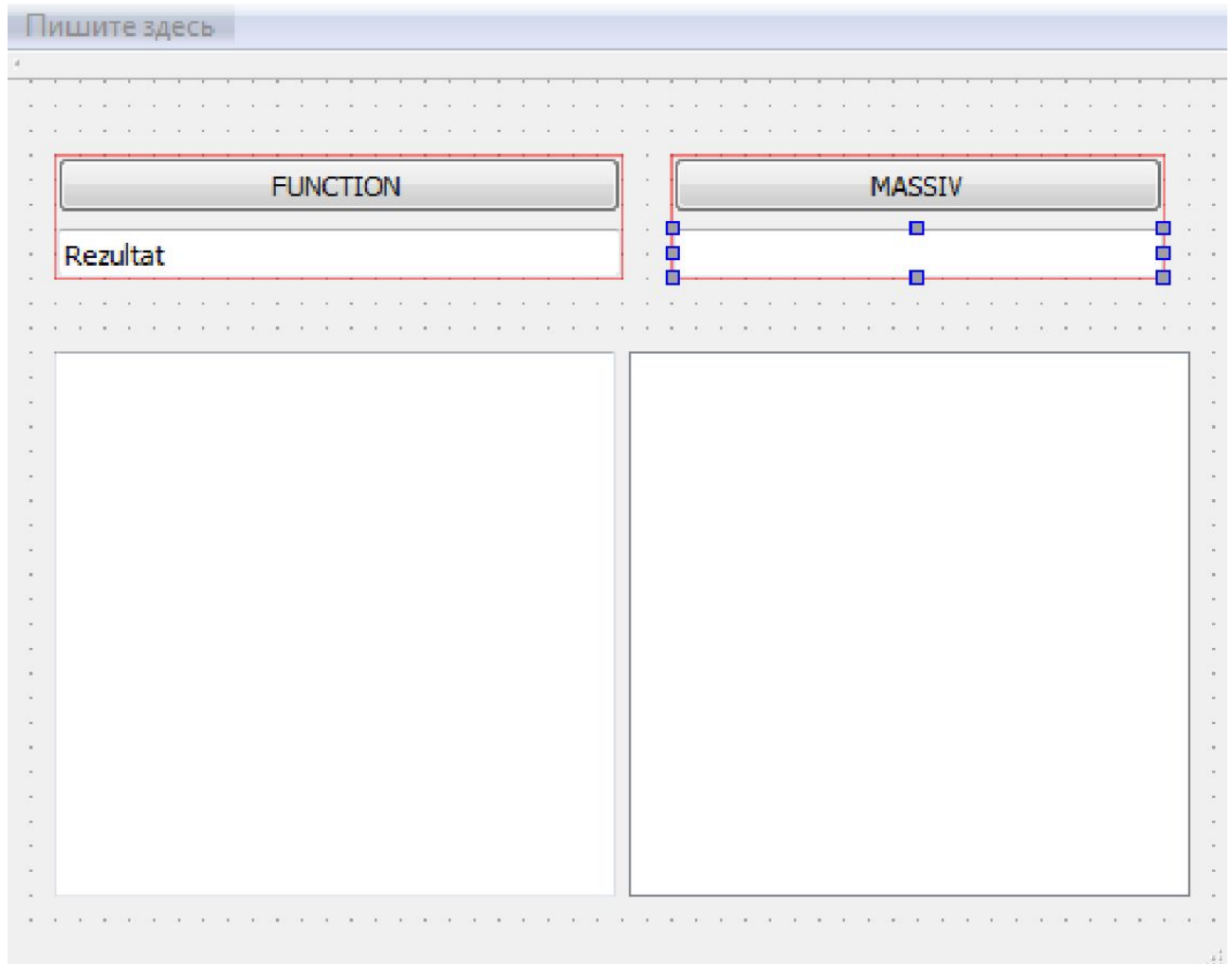
**Настроить свойства виджетов!**

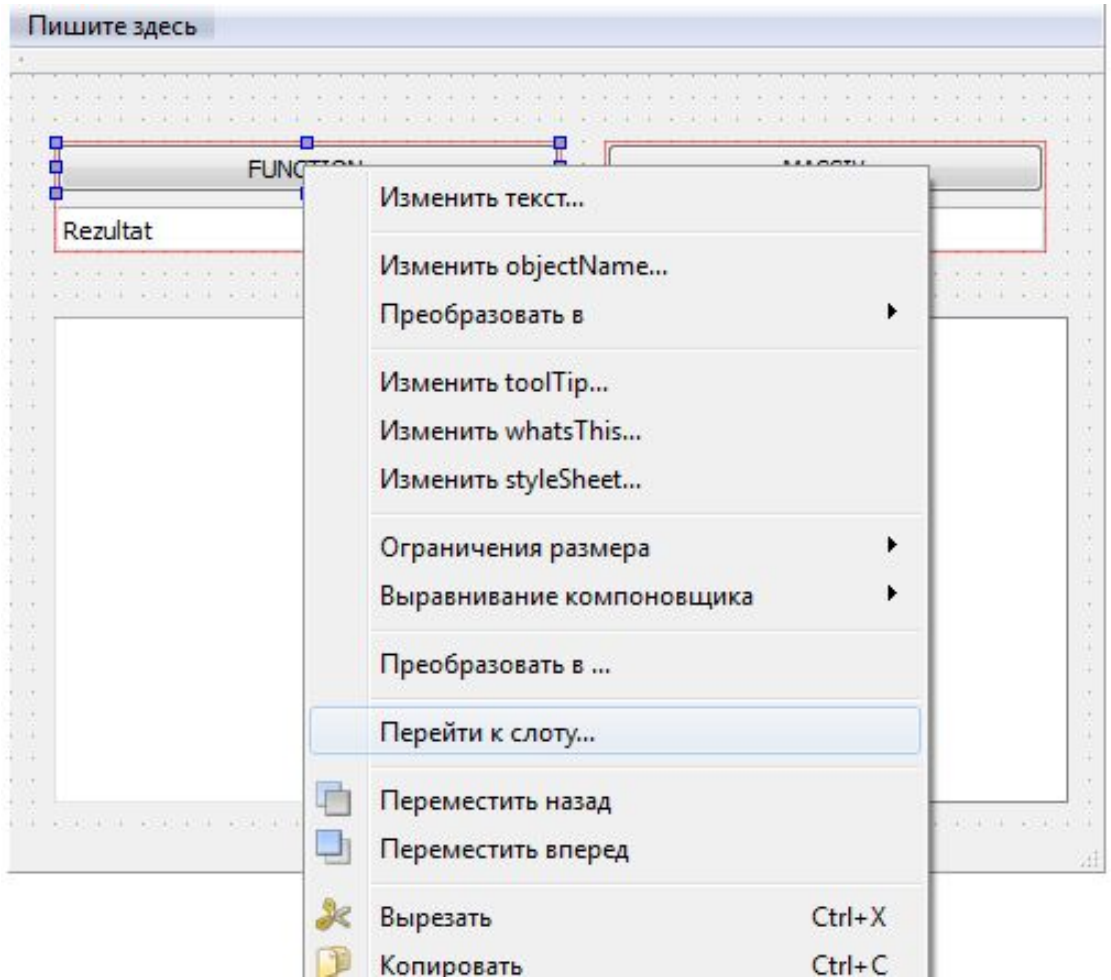
# Файл проекта

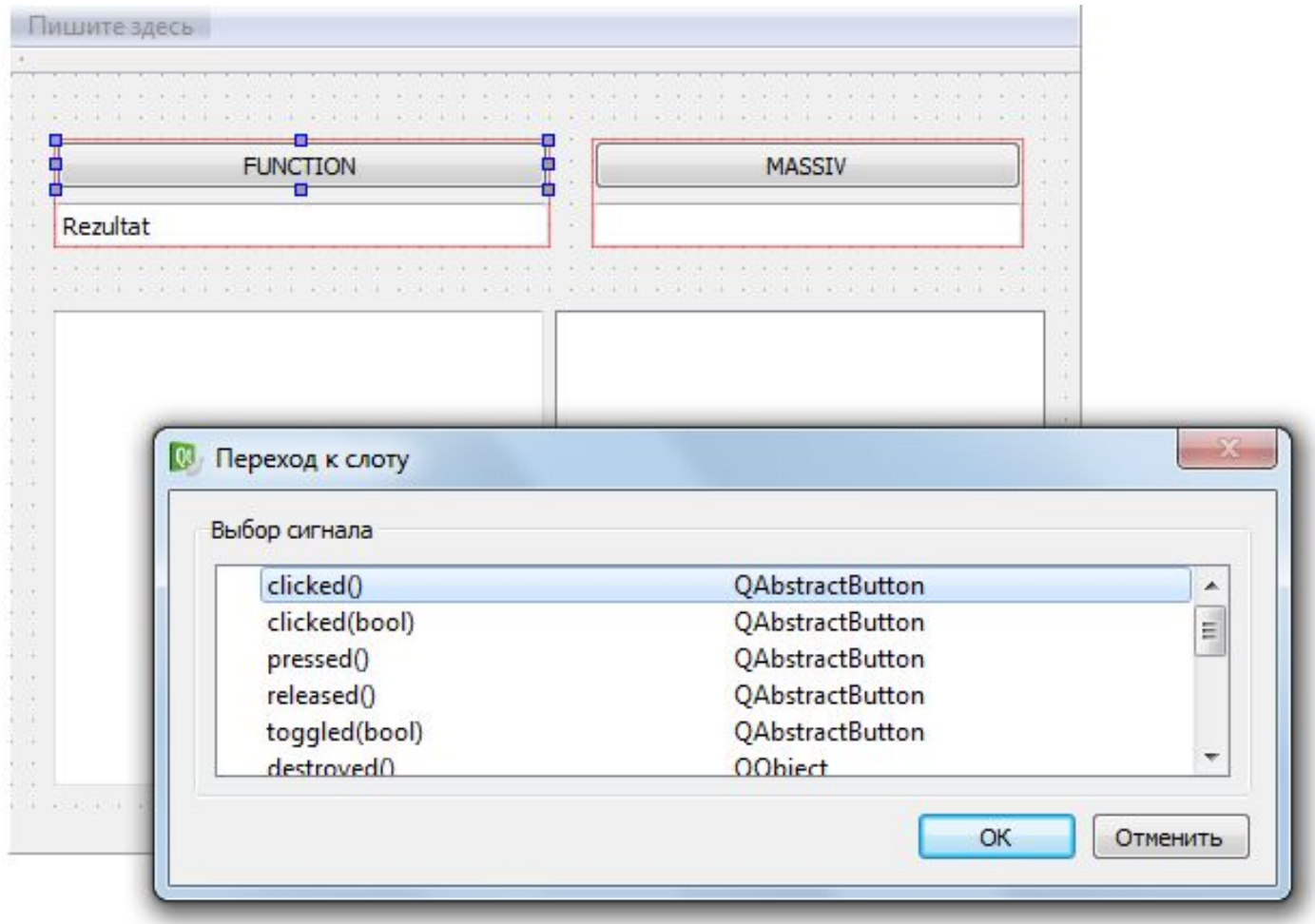
ZF_Laba.pro

```
#-------------------------------------------------
#
# Project created by QtCreator 2017-01-24T15:22:13
#
#-------------------------------------------------

QT       += core gui

greaterThan(QT_MAJOR_VERSION, 4): QT += widgets

TARGET = ZF_Laba
TEMPLATE = app


SOURCES += main.cpp\
        mainwindow.cpp

HEADERS  += mainwindow.h

FORMS    += mainwindow.ui

```


Пишите здесь
FUNCTION
Rezultat
Инспектор объектов
Объект
Класс
MainWindow
QMainWindow
centralWidget
QWidget
vert...yout
QVBoxLayout
lineEdit
QLineEdit
pushButton
QPushButton
splitter
QSplitter
tabl...dget
QTableWidget
textEdit
QTextEdit
menuBar
QMenuBar
mainToolBar
QToolBar
statusBar
QStatusBar
Редактор свойств
Фильтр
lineEdit : QLineEdit
Свойство
Значение
text
Rezultat
maxLength
32767
frame
echoMode
Normal
cursorPosition
8
alignment
AlignLeft, AlignV...
dragEnabled
Редактор сигналов и слотов
Отправитель
Сигнал
Получатель
Слот


Пишите здесь
FUNCTION
MASSIV
Rezultat


Пишите здесь
Rezultat
Изменить текст...
Изменить objectName...
Преобразовать в
Изменить toolTip...
Изменить whatsThis...
Изменить styleSheet...
Ограничения размера
Выравнивание компоновщика
Преобразовать в ...
Перейти к слоту...
Переместить назад
Переместить вперед
Вырезать Ctrl+X
Копировать Ctrl+C

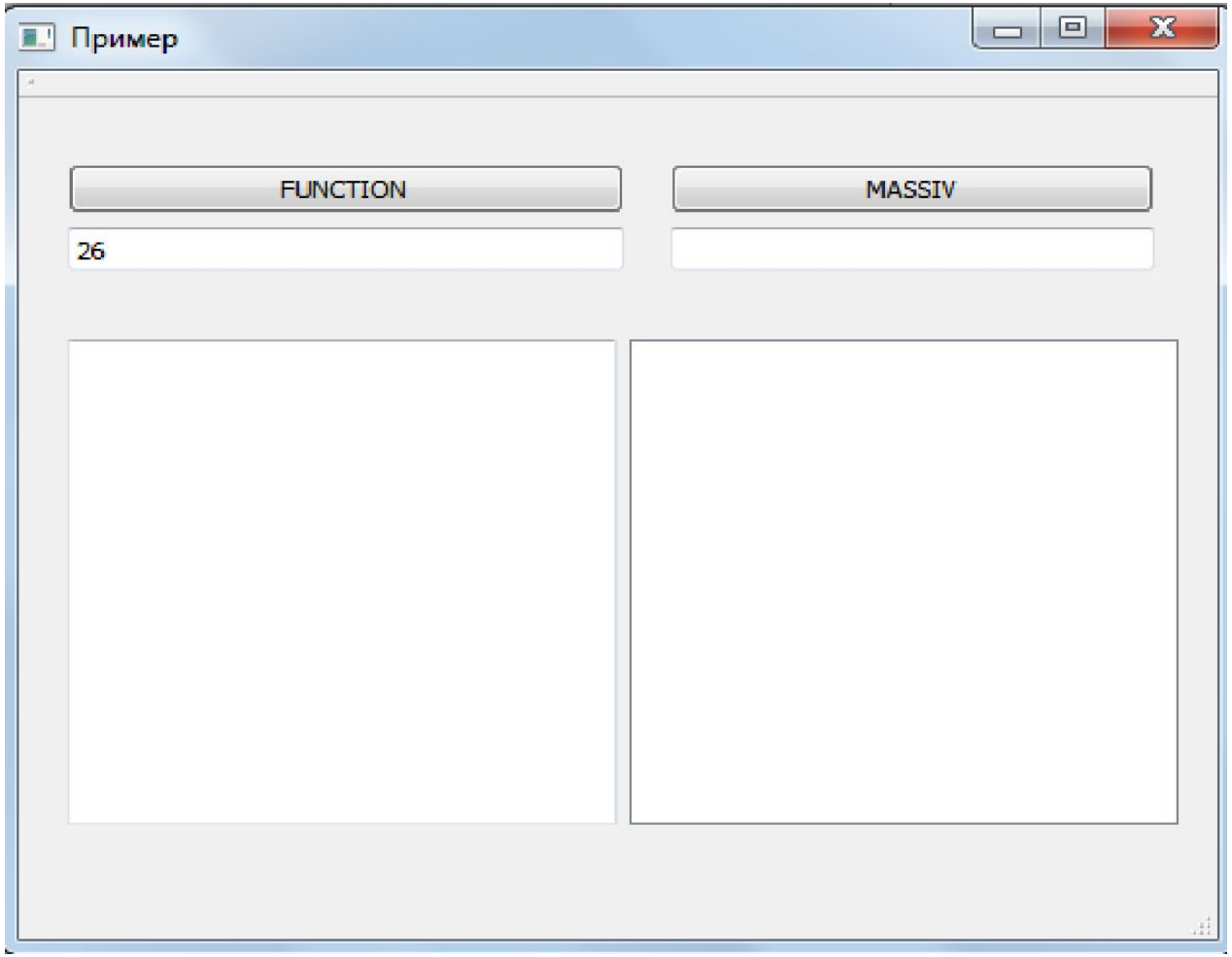

Пишите здесь
FUNCTION
MASSIV
Rezultat
Переход к слоту
Выбор сигнала
clicked() QAbstractButton
clicked(bool) QAbstractButton
pressed() QAbstractButton
released() QAbstractButton
toggled(bool) QAbstractButton
destroyed() QObject
OK
Отменить

main.cpp*

```
1   #include "mainwindow.h"
2   #include <QApplication>
3
4   int main(int argc, char *argv[])
5   {
6       QApplication a(argc, argv);
7       MainWindow w;
8       w.setWindowTitle("Пример");
9       w.show();
10
11      return a.exec();
12  }
13
```

```
#ifndef MAINWINDOW_H
#define MAINWINDOW_H
#include <QMainWindow>
#include <QLabel>
#include <QLineEdit>
namespace Ui {
class MainWindow;
}
class MainWindow : public QMainWindow
{
    Q_OBJECT

public:
    explicit MainWindow(QWidget *parent = 0);
    ~MainWindow();

private slots:
    void on_pushButton_clicked();

private:
    Ui::MainWindow *ui;
    QLineEdit *lineText;
};

#endif // MAINWINDOW_H
```

```
16  void MainWindow::on_pushButton_clicked()
17  {    int i;
18       lineText=ui->lineEdit;
19       QString str="";
20       for(i=1;i<=5;i++)
21       {
22           i=i*i;
23       }
24       str.append(QString("%1").arg(i));
25
26       ui->lineEdit->setText(str);
27  }
28
```


Пример
FUNCTION
MASSIV
26

# Добавим

```
#include <QTextEdit>
```

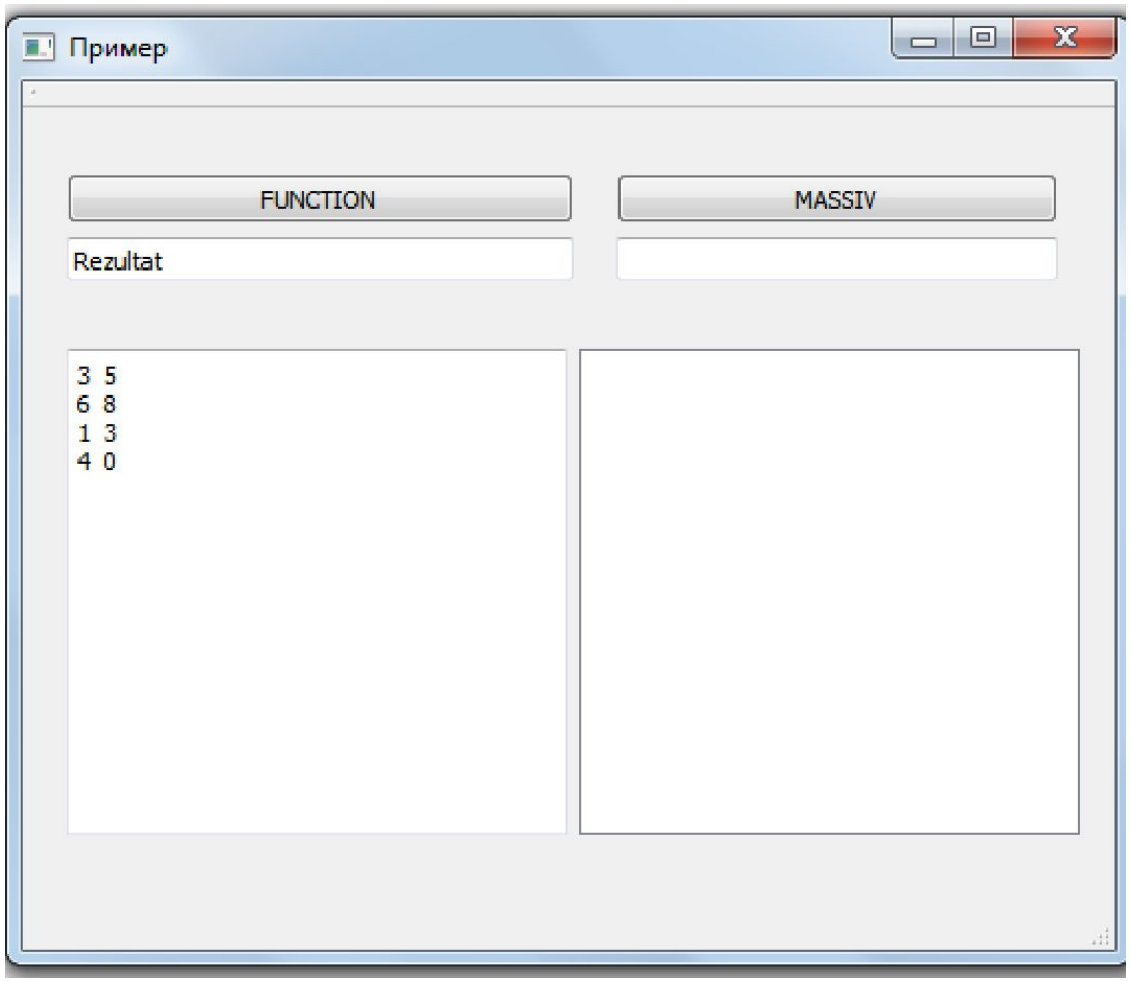
И

```
private: Ui::MainWindow *ui;
QLineEdit *lineText;
QTextEdit *textEdit;
```

# В файле реализации

```
29  void MainWindow::on_pushButton_2_clicked()
30  {
31      QString str;
32      int a[4][2]={{3,5},{6,8},{1,3},{4,0}};
33      for( int i = 0; i <4; i++ )
34      {
35      for( int j = 0; j <2; j++ )
36      str.append(QString("%1  ").arg(a[i][j]));
37      str.append(QString("\n"));
38      }
39      ui->textEdit->setText(str);
40  }
```

# Результат



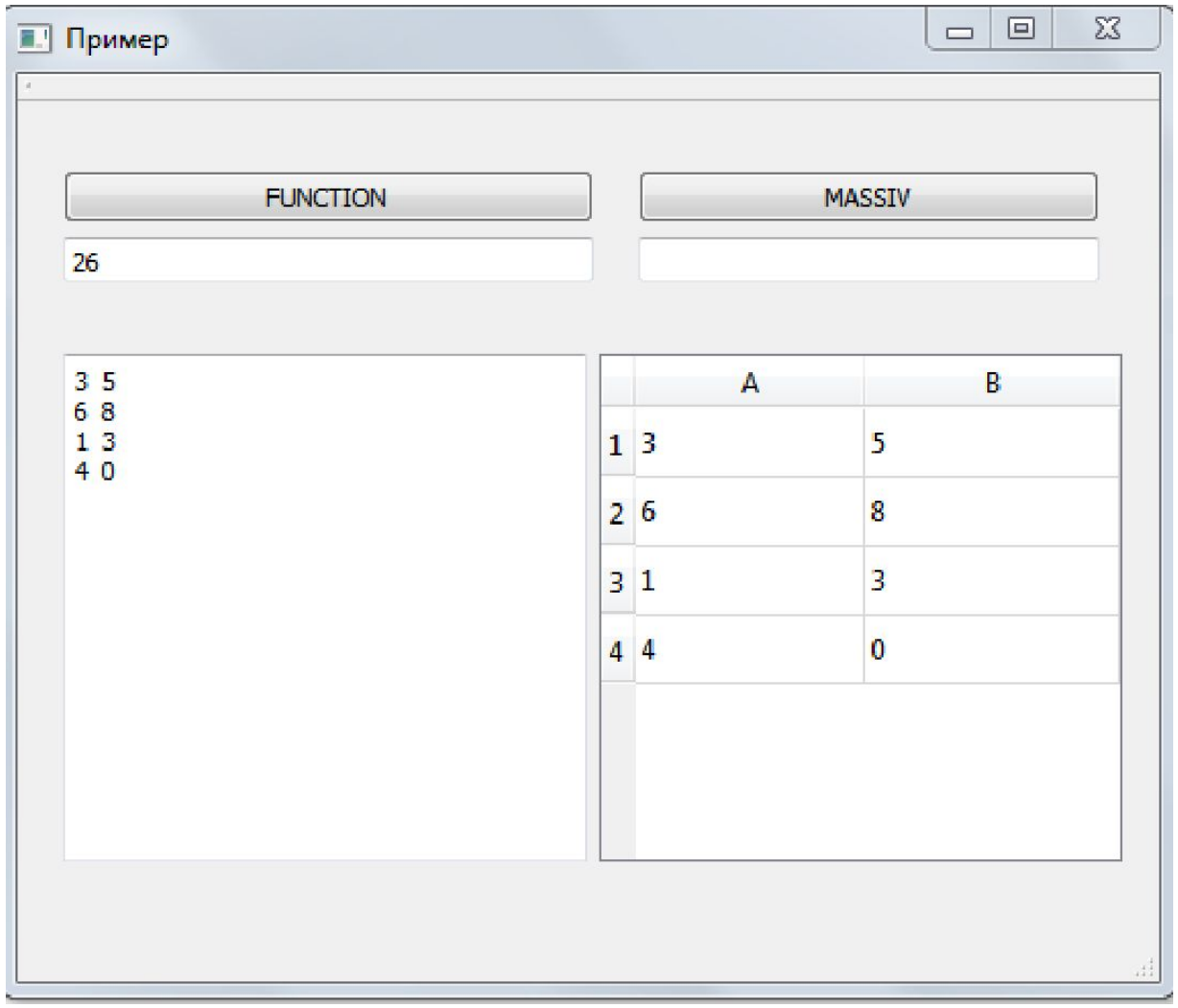
# Добавим вывод матрицы в таблицу

```
QString str;
int a[4][2]={{3,5},{6,8},{1,3},{4,0}};
QFont font; ui->tableWidget->setColumnCount(2);
// Указываем число строк и столбцов
ui->tableWidget->setRowCount(4);
// Включаем сетку
ui->tableWidget->setShowGrid(true);
ui->tableWidget->setHorizontalHeaderLabels(QStringList() << tr("A") << tr("B"));
// Разрешаем выделение только одного элемента
ui->tableWidget->setSelectionMode (QAbstractItemView::SingleSelection);
// Разрешаем выделение построчно
ui->tableWidget-> setSelectionBehavior (QAbstractItemView::SelectRows);
// Растягиваем последнюю колонку на всё доступное пространство
ui->tableWidget->horizontalHeader()->setStretchLastSection(true);
```

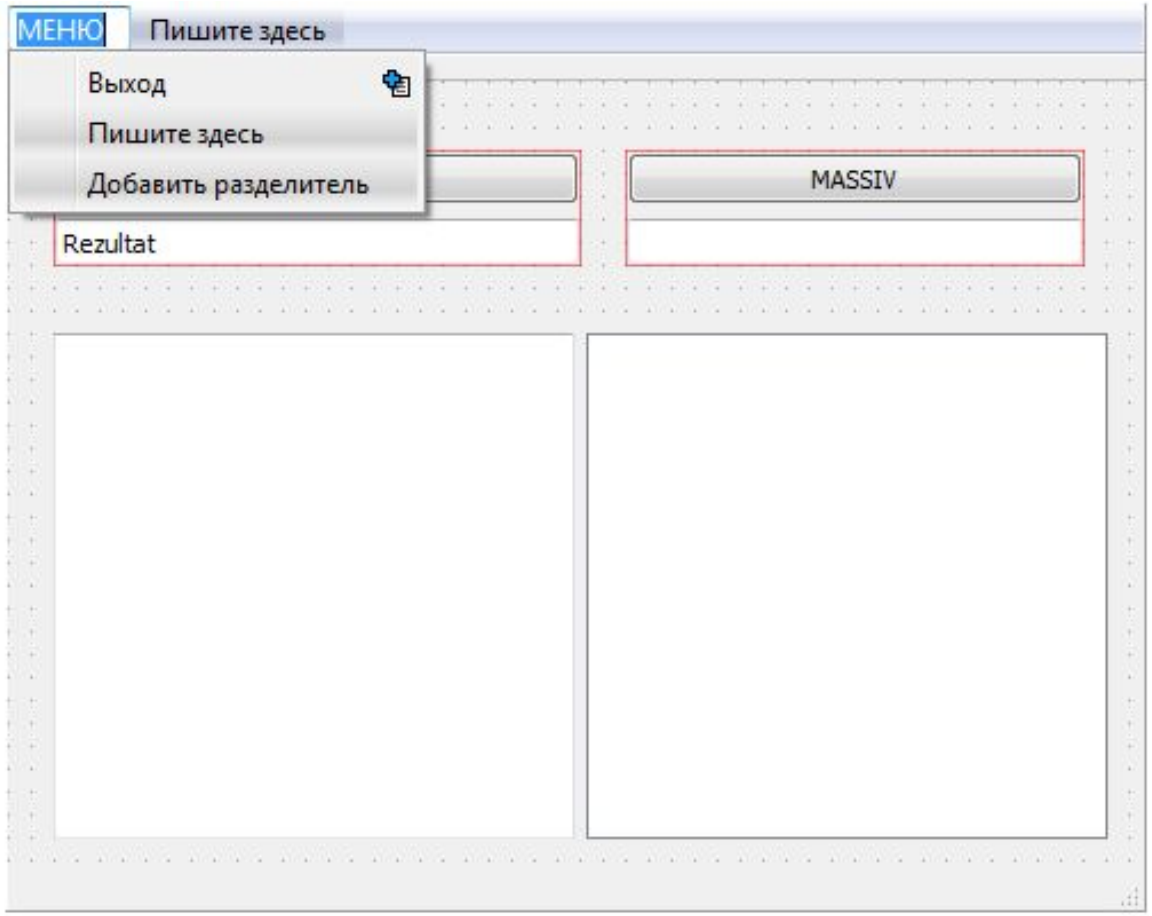
# продолжение

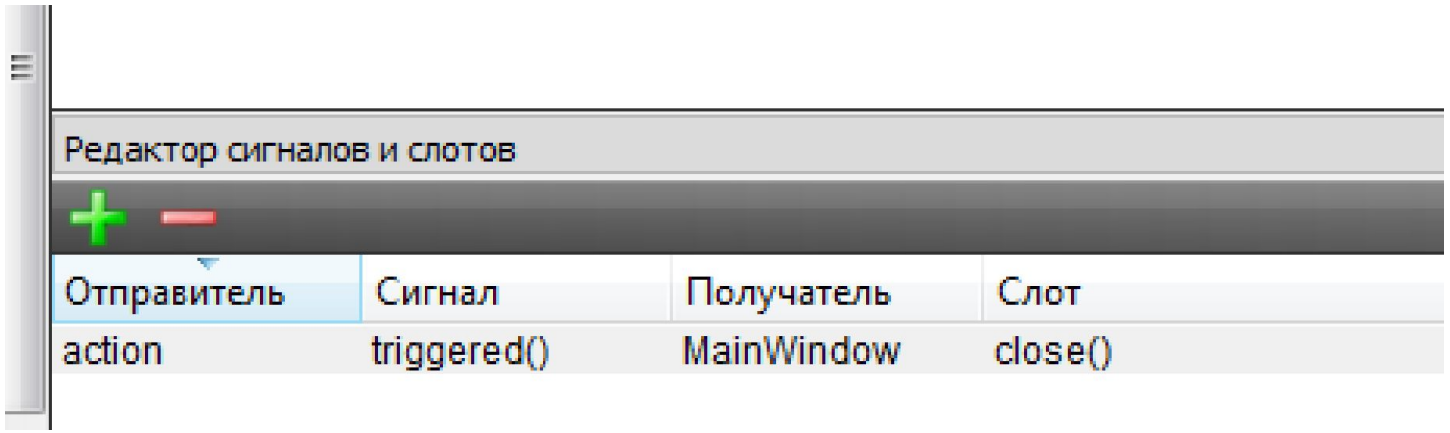
```
for( int i = 0; i <4; i++ )
{ for( int j = 0; j <2; j++ )
{
str.append(QString("%1 ").arg(a[i][j]));
// Добавим вывод в таблицу
QTableWidgetItem *item=new QTableWidgetItem;
item->setText(QString::number(a[i][j]));
ui->tableWidget->setItem(i, j, item); }
str.append(QString("\n"));
 }
ui->textEdit->setText(str);
```

# Конечный результат

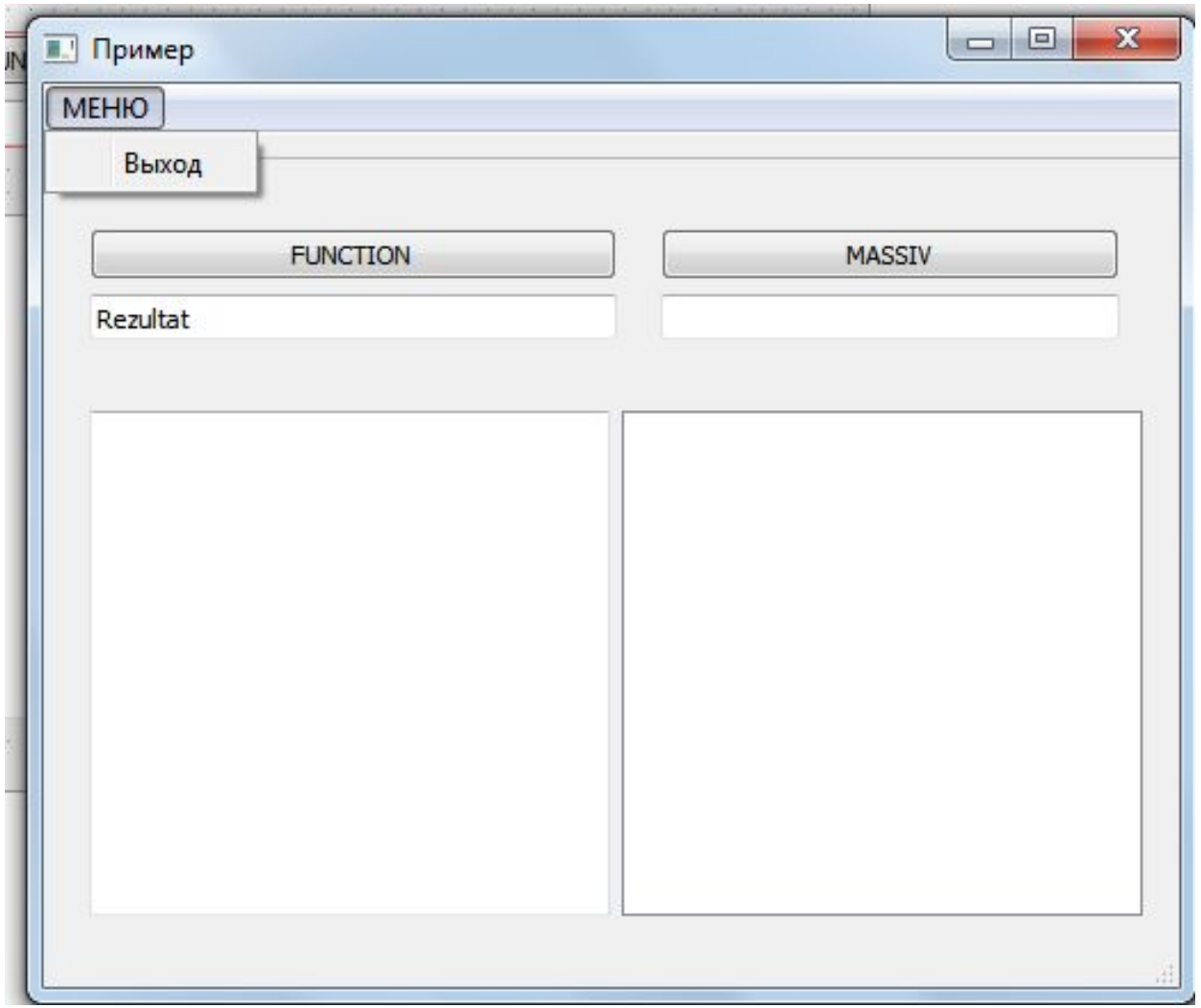


# Добавим меню

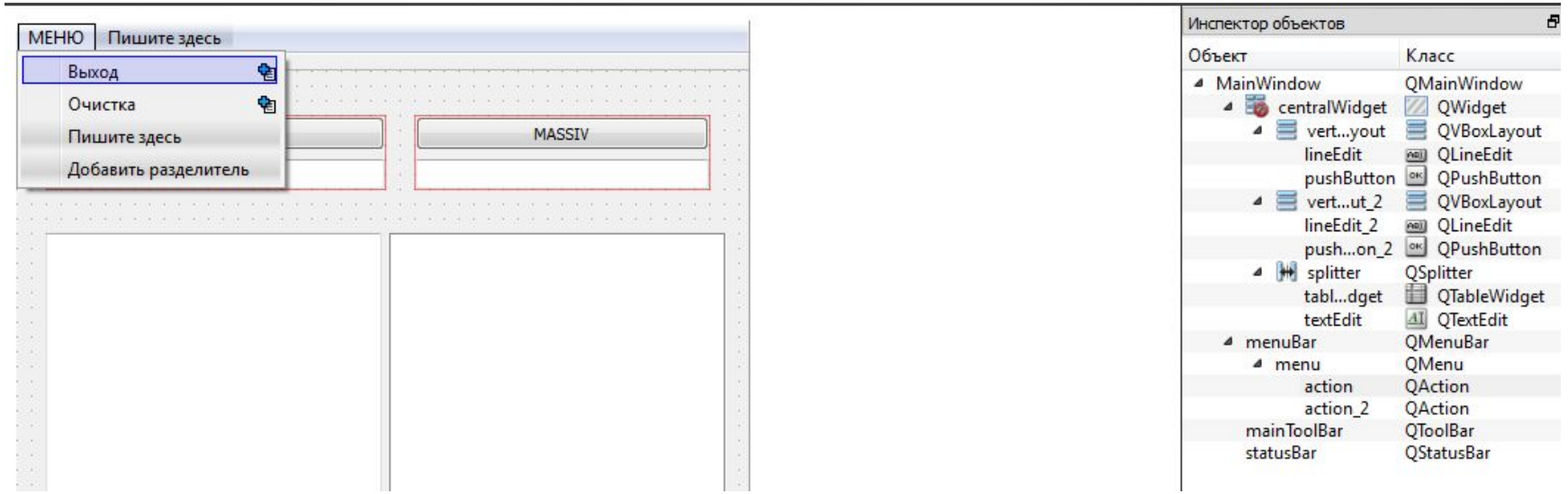


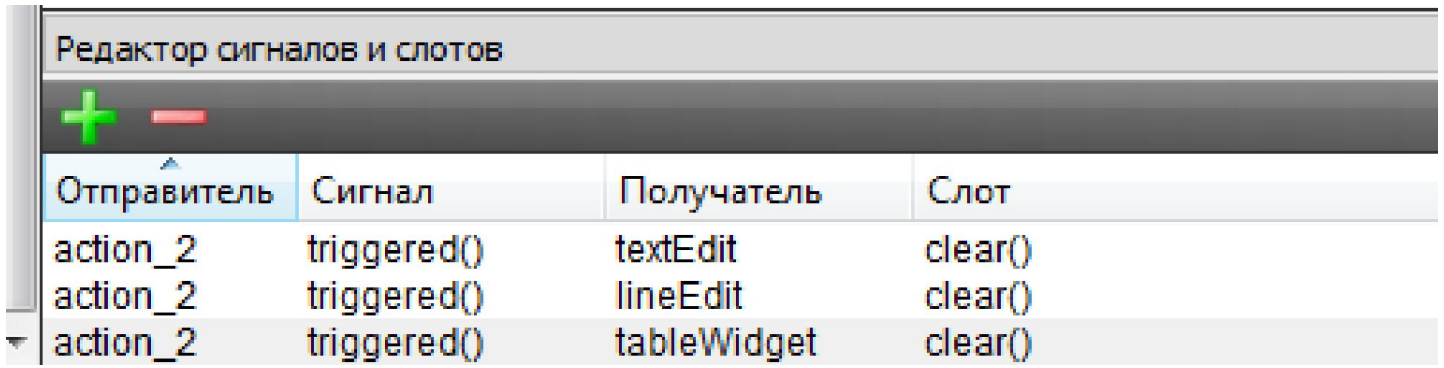

Редактор сигналов и слотов
Отправитель
Сигнал
Получатель
Слот
action
triggered()
MainWindow
close()

# Результат



# Добавим пункт меню «Очистка»




Редактор сигналов и слотов
Отправитель
Сигнал
Получатель
Слот
action_2
triggered()
textEdit
clear()
action_2
triggered()
lineEdit
clear()
action_2
triggered()
tableWidget
clear()