# Задание 1

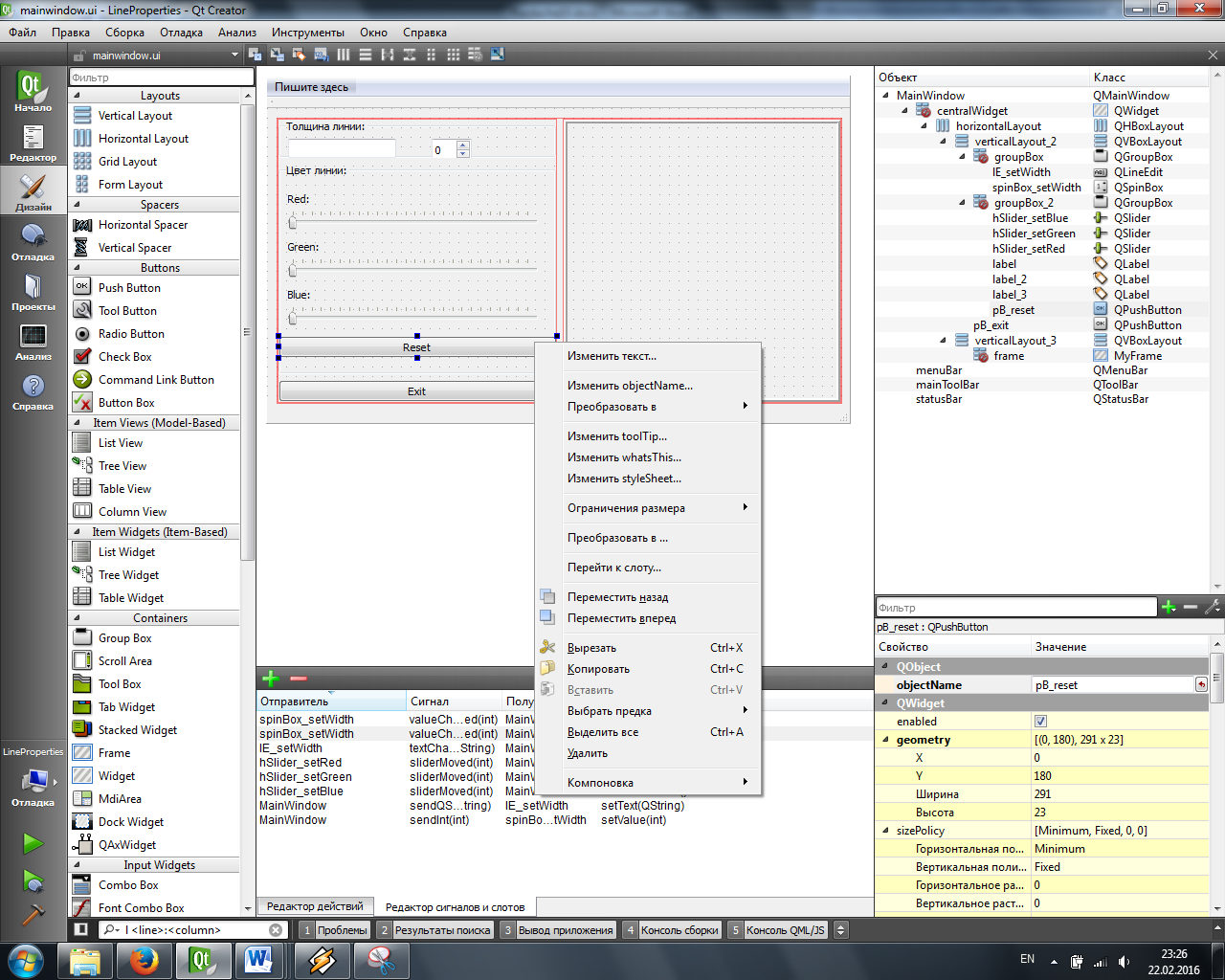
* Реализуйте с помощью дизайнера связь QLineEdit и QSpinBox – реакцию QSpinBox на изменение значения QLineEdit.
* Создайте связь отображающую изменение толщины линии в зависимости значения QSpinBox.

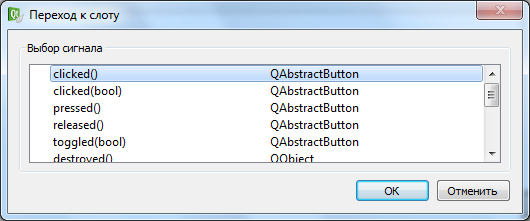
Подсказка:

* + Соедините сигнал QSpinBox valueChanged(int) и слот redefineValues();
  + Для создания объекта QPen воспользуетесь следующим конструктором:  [**QPen**](qthelp://org.qt-project.qtgui.532/qtgui/qpen.html#QPen-4)(const QBrush &*brush*, qreal*width*, Qt::PenStyle*style* = Qt::SolidLine, Qt::PenCapStyle*cap* = Qt::SquareCap, Qt::PenJoinStyle*join* = Qt::BevelJoin). Достаточно использовать только два первых параметра
* Предусмотрите сброс всех параметров к изначальным при нажатии кнопки Reset

Подсказка:

* + Параметры по умолчанию устанавливаются при помощи метода setValue()
  + С помощью дизайнера можно создать слот-обработчик этого события следующим образом:





* + Ок

# Задание2

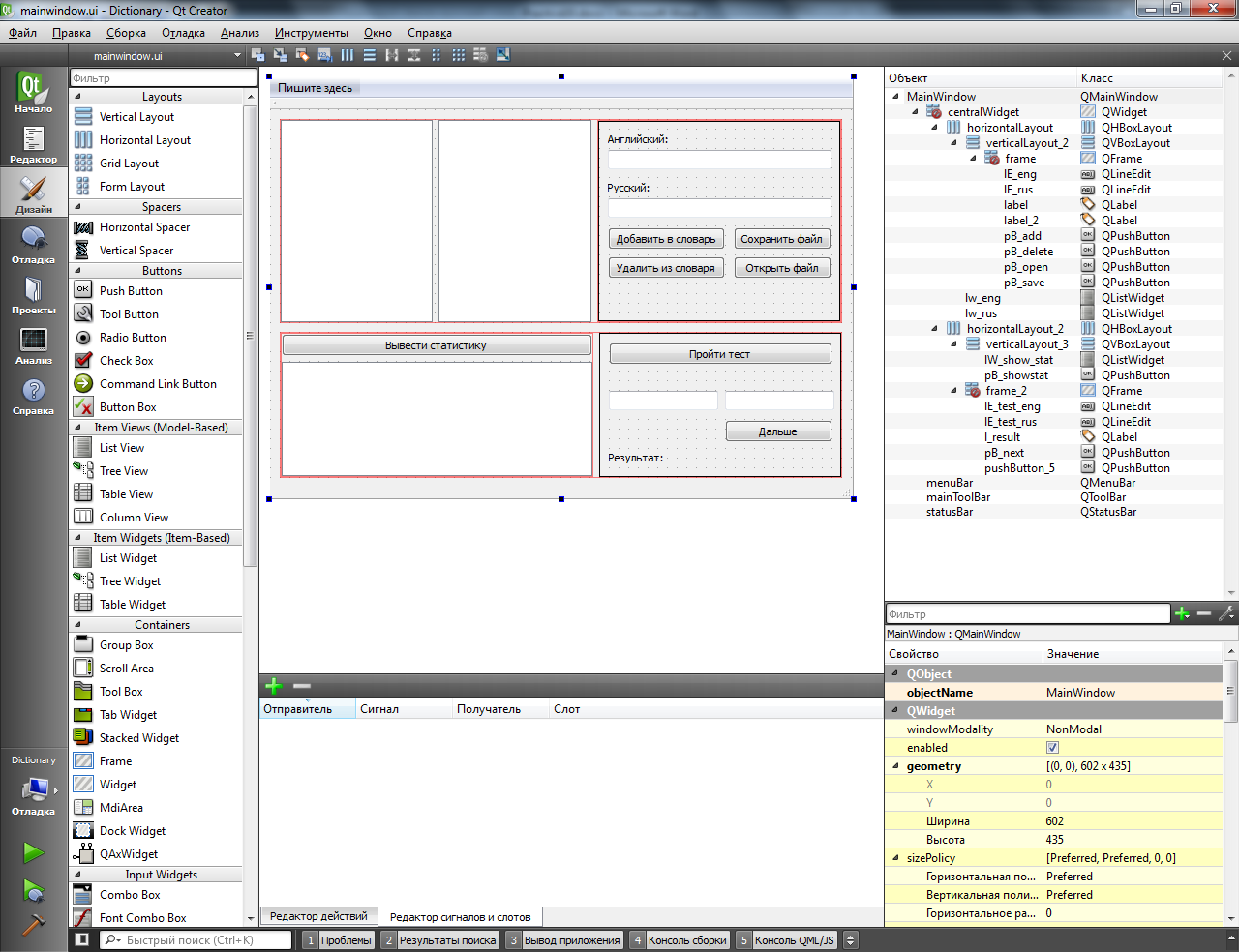
Реализовать приложение «Словарь».

Наш словарь состоит 5-ти пар слов. Каждая пара связывает одно английское и одно русское слово. Каждое слово уникально.

Функции словаря:

* Добавление пары;
* Удаление последней пары;
* Просмотр всех пар;
* Вывод статистики по имеющимся карточкам в словаре;
  + Количество слов в словаре;
  + Средняя длина английского слова;
  + Средняя длина русского слова;
  + Количество букв во всех словах словаря;
* Сохранение словаря в файл txt и загрузка его из файла;
* Проверка на знание словаря.
  + При запуске тестирования программа предлагает 5 английских слов из словаря в случайном порядке;
  + Пользователь должен записать его пару на русском языке;
  + После завершения тестирования выводится результат – количество правильных ответов в процентах.

### Внешний вид формы



QListWidget 3

QListWidget 2

QListWidget 1

QLabel

QPushButton

QLineEdit

QLineEdit

QPushButton

QPushButton

QFrame

Вертикальная компоновка

QFrame

QPushButton

QPushButton

QPushButton

QPushButton

QLineEdit

QLabel

QLineEdit

QLabel

Горизонтальная компоновка

Горизонтальная компоновка

Центральный виджет

### Описание работы программы

* Для того, чтобы добавить пару пользователь должен английский и русский варианты в lineEditы верхнего QFrame и нажать кнопку «Добавить в словарь»
* Удалять можно только последнюю добавленную пару.
* Кнопка «Дальше» - предлагает следующее английское слово в тесте.
* В QListWidget1 отображается список английских слов.
* В QListWidget2 отображается список русских слов.
* Порядковые номера английских слов и их пары совпадают.

### Некоторые элементы архитектуры программы

* Пара слов (англ./рус.) представляет собой объект класса Pair, содержащий 2 объекта QString.
* Словарь Dictionary хранит статический массив указателей класса Pair m\_pair.
* Размер массива задайте самостоятельно.
* Так как все основные действия со словарем «происходят» в классе главного окна, логично объявить в классе указатель на Dictionary m\_dict.
* Подумайте, что нужно сделать, чтобы добавление/удаление объекта происходило в нужную позицию массива.
* Не забудьте предусмотреть защиту от переполнения/удаления из пустого массива.
* Функции для работы с QListWidget:
  + void addItem(const QString & label) – добавление элемента в конец списка;
  + QListWidgetItem \* takeItem(int row) – удаление элемента, расположенного на строке row;
  + void clear() – очистка содержимого QListWidget.
* Желательно с помощью фунцкии void QPushButton::setEnabled(bool) регулировать возможность нажатия кнопок.