

## Создание программы с помощью PyQt

*Азаров Андрей Евгеньевич*

*Приамурский государственный университет им. Шолом-Алейхема*

*студент*

### **Аннотация**

Данное исследование содержит информацию о создании программы при помощи pyqt. Создание программы проходит в несколько этапов: создание макета дизайна, написание кода, компиляция. В результате имеется готовая к эксплуатации программа.

**Ключевые слова:** python, pyqt, фреймворк.

## Creating a program with PyQt

*Azarov Andrey Evgenevich*

*Sholom-Aleichem Priamursky State University*

*student*

### **Abstract**

This study contains information about creating programs using pyqt. Creating programs takes place in several stages: the design layout, writing code, compilation. As a result, there is a ready-to-use program.

**Keywords:** python, pyqt, framework.

Ещё два десятка лет назад, программы создавались в простых средах разработки, которые внешне напоминали обычный блокнот. Для разработки современного программного обеспечения также некоторые среды выглядят как обычный блокнот, однако, в большинстве случаев, среда разработки программного обеспечения имеет графический интерфейс, который ускоряет разработку любой программы в десятки раз и экономит время программистам, а также уменьшает конечную стоимость программ.

Стандартных средств создания графического интерфейса от разработчиков языка нет и причин этому много. Для создания программ с графическим интерфейсом используются различные фреймворки. Фреймворк – это набор сторонних библиотек, включающий в себя различные шаблоны облегчающее разработку и объединение разных модулей программного проекта.

Фреймворк PyQt — это набор библиотек графического фреймворка Qt для языка программирования Python, выполненный в виде расширения Python. Qt — это кроссплатформенный фреймворк для разработки программного обеспечения на языке программирования C++. И так как стандартный язык для работы с Qt является язык C++, было создано

ответвление в виде фреймворка PyQt для работы с данным фреймворком на языке python.

Данное исследование проводится с целью изучения возможностей фреймворка PyQt по созданию программ.

Python имеет множество различных фреймворков для создания графического интерфейса в программах. Одним из популярных фреймворков является kivy [1], с помощью kivy можно создавать интерфейс программ на мобильные платформы, такие как android и ios. Libavg [2] библиотека написанная на C++ с помощью которой пишутся интерфейсы для ОС Linux, Mac OS X и Windows, используется не только для создания простых программ, но и для компьютерных игр. PyGUI [3] фреймворк для создания интерфейса на Unix платформах.

В данном исследовании рассматривается фреймворк PyQt. Фреймворк выбран из-за своей популярности среди разработчиков. О данном программном средстве написано множество статей. Статья 2011 года Python 3 и pyqt [4] описан базовый синтаксис языка Python: типы данных, операторы, условия, циклы, регулярные выражения, встроенные функции. В статье А.А. Юлаева [5] рассматриваются возможности языка программирования Python в сочетании со сторонними модулями PyQt и PyVISA применительно к автоматизации процессов межоперационного и выходного контроля параметров арсенид-галлиевых (GaAs) структур и монолитных интегральных схем (МИС) на их основе, демонстрирую, что с помощью pyqt можно создать сложные информационные системы. Зарубежное сообщество также проводило исследования и изучало pyqt. Одним из зарубежных примеров является статья python's pyqt toolkit [6] в журнале Dr. Dobb's journal рассматривающая возможности фреймворка, классы и связь с C++.

Для создания программы с графическим интерфейсом с помощью PyQt необходимо скачать PyQt Designer — средство для проектирования форм графического интерфейса (далее GUI англ. – graphical user interface) и любую среду разработки (далее IDE англ. – Integrated Development Environment). Одной из популярных и бесплатных IDE для python является Visual Studio Community.

В первую очередь необходимо запустить Qt Designer который имеет стандартный набор GUI элементов, таких как: кнопки, текстовые поля, чекбоксы, обычный текст, изображения, прогресс бары, вертикальные и горизонтальные полосы прокрутки и многое другое. С помощью технологии drag&drop переносим на форму программы элементы управления: кнопки, текстовые поля, текстовые надписи. Вид созданной будущей программы изображен на рисунке 1.

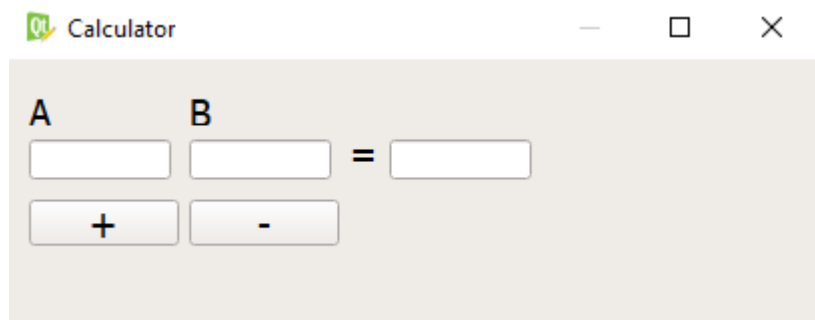


Рисунок 1. GUI созданный с помощью QtDesigner

После создания GUI программы с помощью QtDesigner в результате имеется файл дизайна с расширением .ui. Данный файл содержит текстовую разметку интерфейса программы на языке xml. Python не поймет расширение .ui, которое было создано для обычного Qt и языка C++. Файл .ui необходимо конвертировать в файл понятный языку программирования python и расширению PyQt. Для конвертирования у расширения PyQt имеется утилита – pyqtdeploy – для использования данной утилиты необходимо открыть командную строку и ввести следующую команду: C:\Python\Scripts\pyuic5.exe -x ui\_main\_window.ui -o ui\_main\_window.py

Где сначала пишем путь к файлу pyuic5, далее имя файла с расширением .ui и режим работы утилиты.

В результате работы данной утилиты получаем файл с расширением .py. Данный файл можно открыть с помощью IDE Visual Studio. Содержимое файла изображено на рисунке 2.

```
</property>
<property name="text">
<string>A</string>
</property>
</widget>
<widget class="QPushButton" name="pushButton_2">
<property name="geometry">
<rect>
<x>90</x>
<y>70</y>
<width>75</width>
<height>23</height>
</rect>
</property>
<property name="font">
<font>
<pointsize>14</pointsize>
</font>
</property>
<property name="text">
<string>-</string>
</property>
</widget>
<widget class="QLabel" name="label_2">
<property name="geometry">
<rect>
<x>90</x>
<y>20</y>
<width>47</width>
<height>13</height>
</rect>
</property>
<property name="font">
```

Рисунок 2. Содержимое файла calculator.py

Теперь появилась возможность обращаться к графическим элементам с помощью кода на языке python.

Напишем следующий код (рисунок 3):

```
def addition(self, a, b):  
    sum=a+b  
    self.text_result.setText(sum)
```

Рисунок 3. Функция addition()

В данном коде написана функция addition() которая принимает два входных параметра – действительные числа a и b, над которыми производится операция сложения. В результате работы функция возвращает сумму чисел. Сумма чисел выводится в текстовое поле результата text\_result.

На данный момент имеет расширение .py и запускать данную программу можно на компьютерах с установленным языком python на компьютере, для запуска программы на любом компьютере под управлением операционной системы windows необходимо файл calculator.py конвертировать с помощью утилиты windeploy. Работа данной утилиты аналогична утилите pyqtdeploy, поэтому подробно она рассматриваться не будет. Программа созданная вышеописанным образом работает на любой версии windows где установлен net.framework 4 версии и выше.

Цели данного исследования были выполнены. Проведено обзорное исследование фреймворка pyqt, изучены основные возможности и написана работоспособная программа. Можно сделать вывод, что фреймворк pyqt современное и актуальное средство для создания пользовательских интерфейсов на языке python. Данное исследование можно применять в учебных целях для пояснения основных нюансов работы с pyqt и языком python.

### Библиографический список

1. Kivy URL: <https://kivy.org/> (дата обращения: 10.01.2019).
2. Libavg URL: <https://www.libavg.de/site/> (дата обращения: 10.01.2019).
3. The Python GUI Project // URL: [http://www.cosc.canterbury.ac.nz/greg.ewing/python\\_gui/](http://www.cosc.canterbury.ac.nz/greg.ewing/python_gui/) (дата обращения: 10.01.2019).
4. Прохоренок Н.А. Python 3 и pyqt. СПб.: БХВ-Петербург, 704.
5. Юлаев А.А. Возможности модулей pyqt и pyvisa для автоматизации процесса измерения параметров gaas структур и миссвч-диапазона на их основе // Материалы конференции. московский институт электроники и математики национального исследовательского университета «Высшая школа экономики». М.: Московский институт электроники и математики НИУ ВШЭ, 2016. С. 110-111.
6. Rempt B. Python'S Pyqt Toolkit // Dr. Dobb's journal: software tools for the professional programmer. 2011. №1.