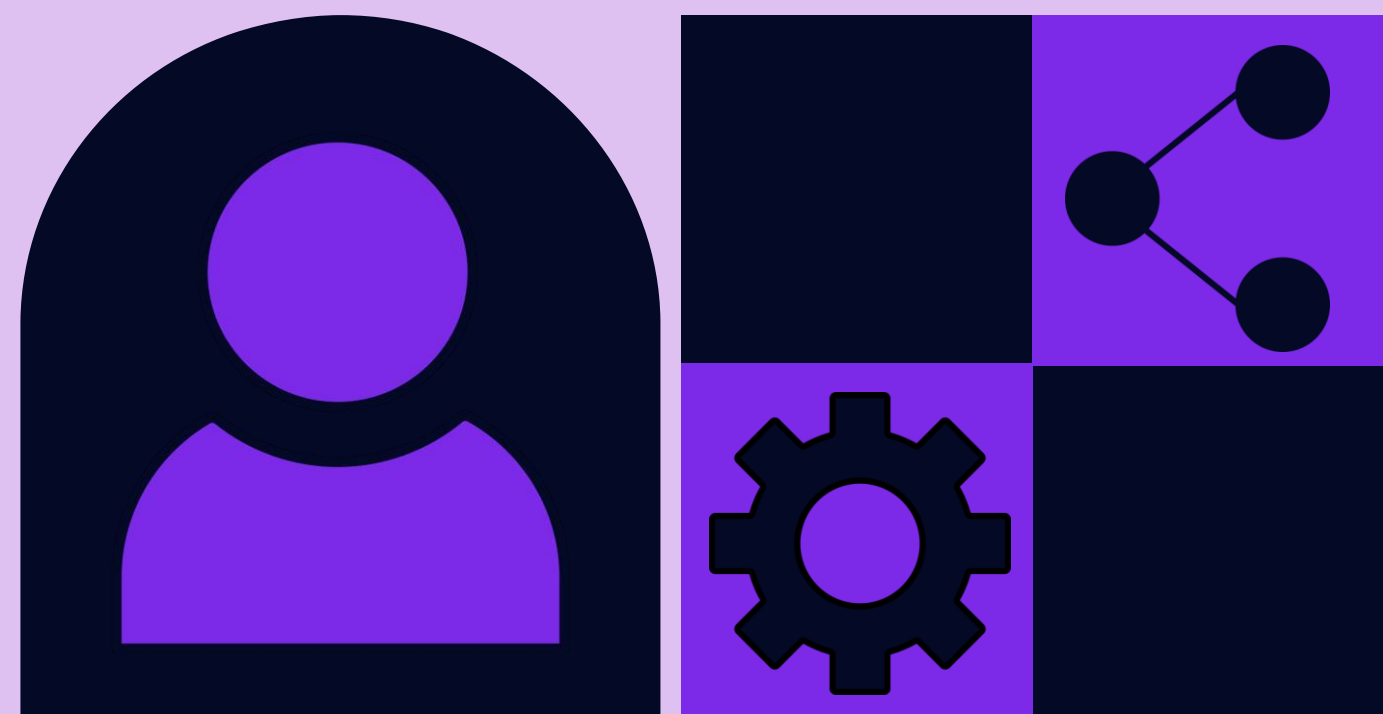


Гришин Илья

# **Data Science.**

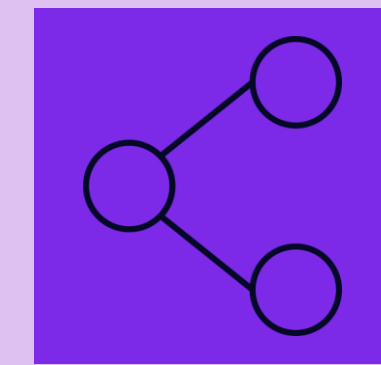
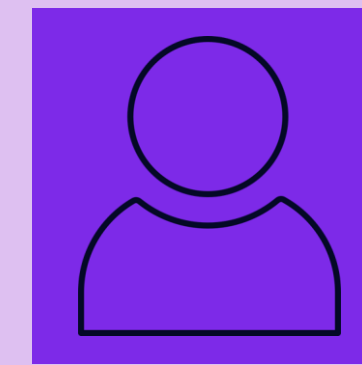
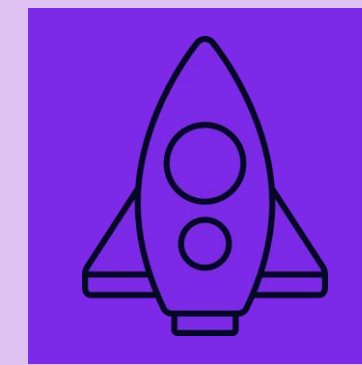
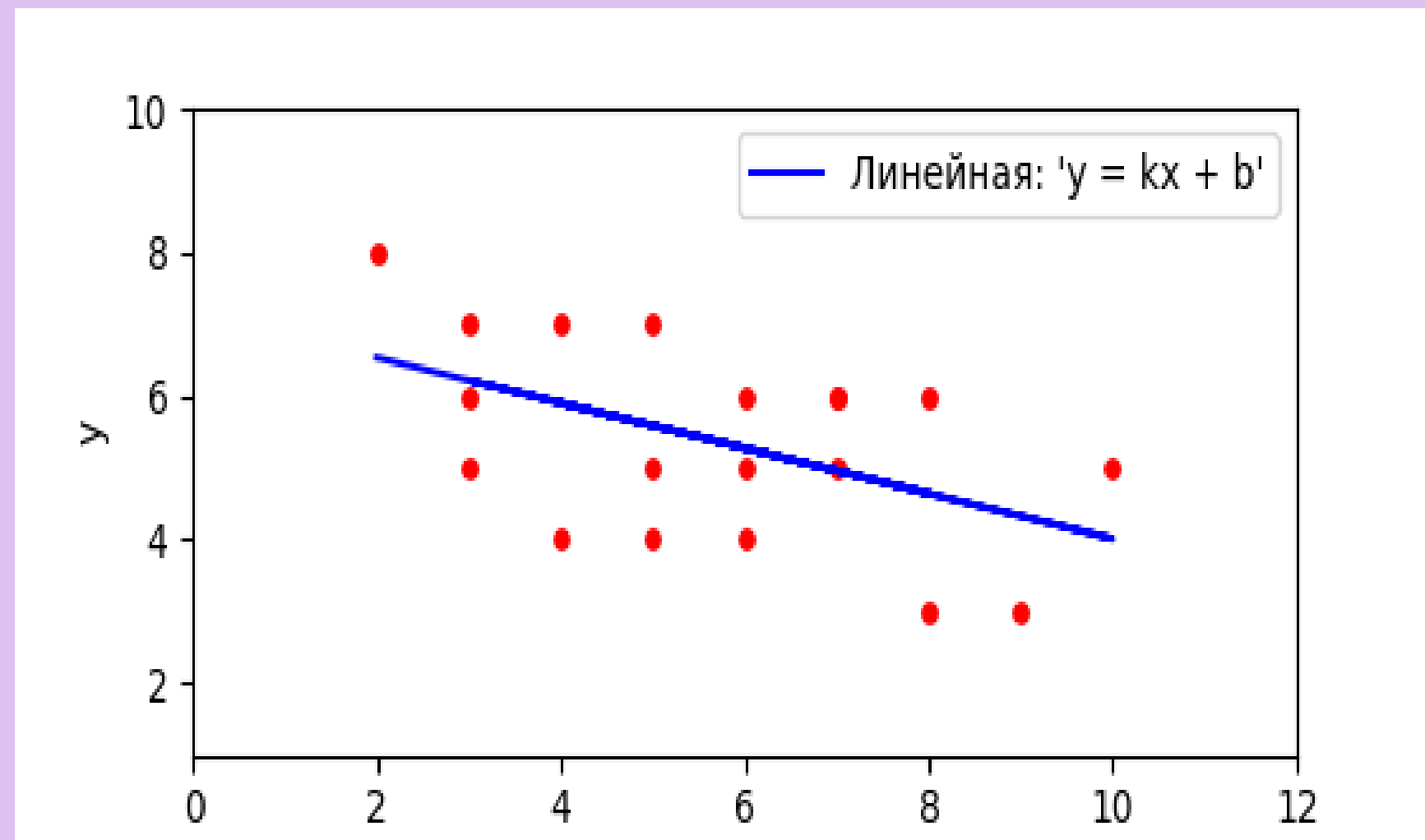
## **PyQt5 program for**

### **function approximation.**



# Что такое аппроксимация?

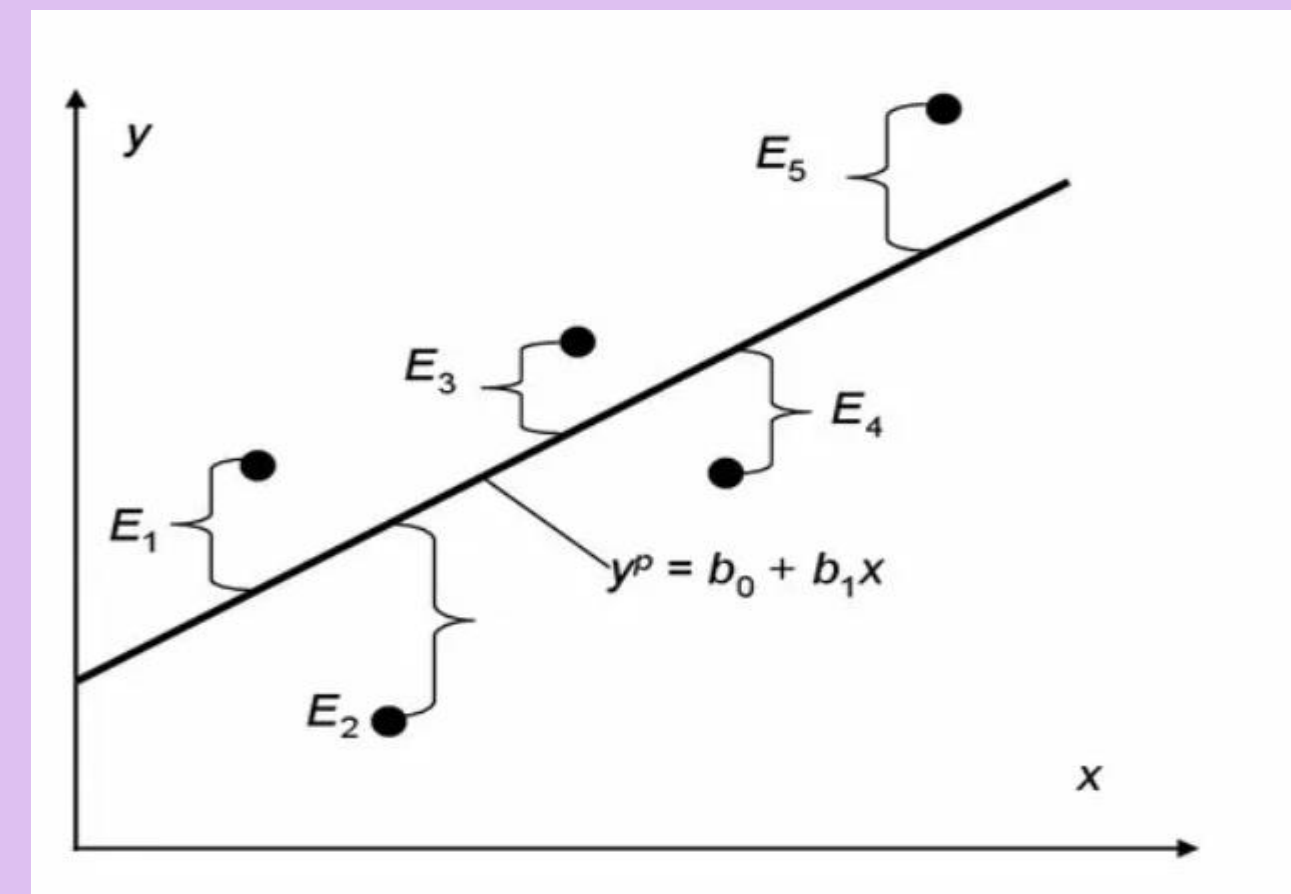
**Аппроксимация – это процесс приближенного представления или замены сложной математической функции, объекта или явления более простой и удобной для анализа моделью. В рамках аппроксимации происходит замена исходного объекта или явления аппроксимирующей функцией или моделью, которая лучше всего описывает его свойства.**



## Каким образом происходит аппроксимация

**Допустим что модель представляет собой прямую линию:  
 $y = b_0 + b_1 \cdot x$**

**Требуется получить такие значения коэффициентов  $b_0$  и  $b_1$ , при которых сумма квадратов  $E$  (расстояний по вертикали от этой точки до прямой линии) является минимальной.**

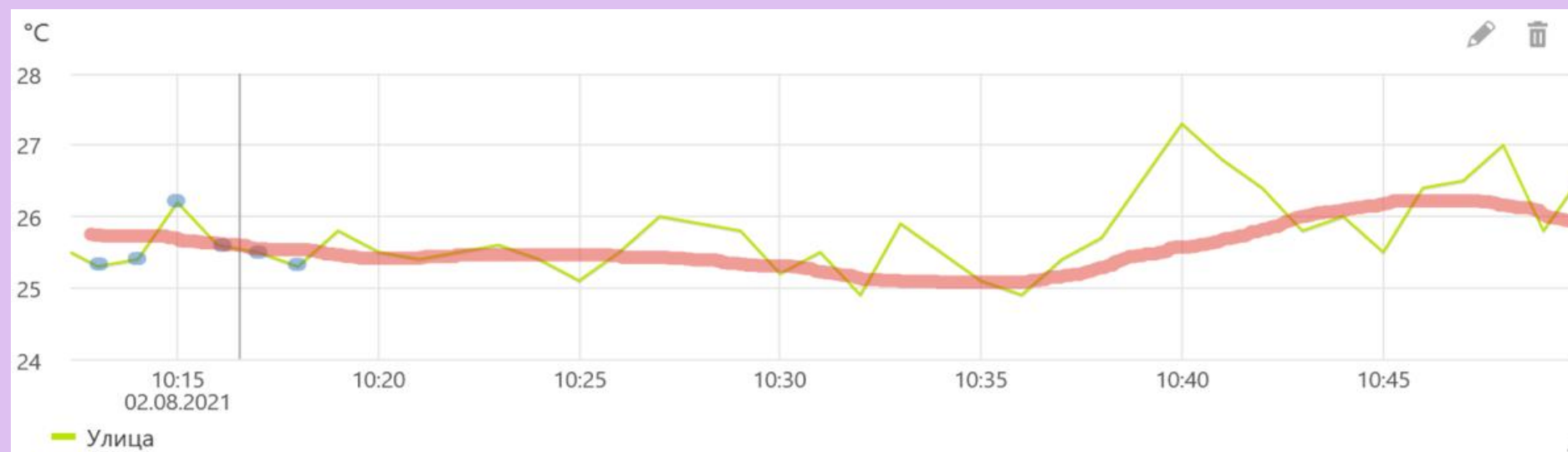


# Идея проекта:

Проект создан для решения задач линейной и нелинейной аппроксимации.

## Для чего нужна аппроксимация?

С помощью аппроксимации можно построить более стабильную и гладкую функцию для дальнейшего изучения. Пример:



Также с помощью аппроксимации функции можно предугадать поведение функции и узнать промежуточные значения.

Она позволяет упростить математические задачи, улучшить точность аналитических моделей и эффективность алгоритмов обработки данных.

Применение аппроксимации широко распространено в научных и инженерных областях.



# Описание реализации

## Элементы PyQt5:

- QTableWidgetItem
- QTextBrowser
- QSpinBox
- QPushButton
- QRadioButton
- QVBoxLayout
- QLabel
- QCheckBox
- QComboBox
- QWidget
- QTabWidget
- QMainWindow
- TableView

## Сторонние функции и библиотеки:

Задача аппроксимации решается с помощью подключения сторонних библиотек таких как *Scipy*, *NumPy*, *mathlib*. Построение графика происходит с помощью функции *'plot()'*.


Подгонка нелинейной аппроксимации с помощью функции *'curve\_fit()'*.

## Возможности:

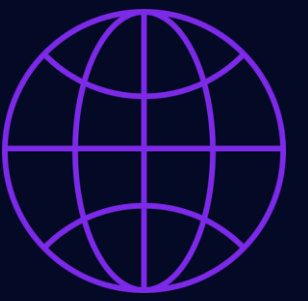
Есть возможность считывания **.XLSX** файлов для удобства ввода координат точек. При помощи библиотеки *openpyxl*.

Также имеется работа с базой данных *SQLite3*.

Есть возможность считывать, изменять, удалять, добавлять элементы.



# Интерфейс



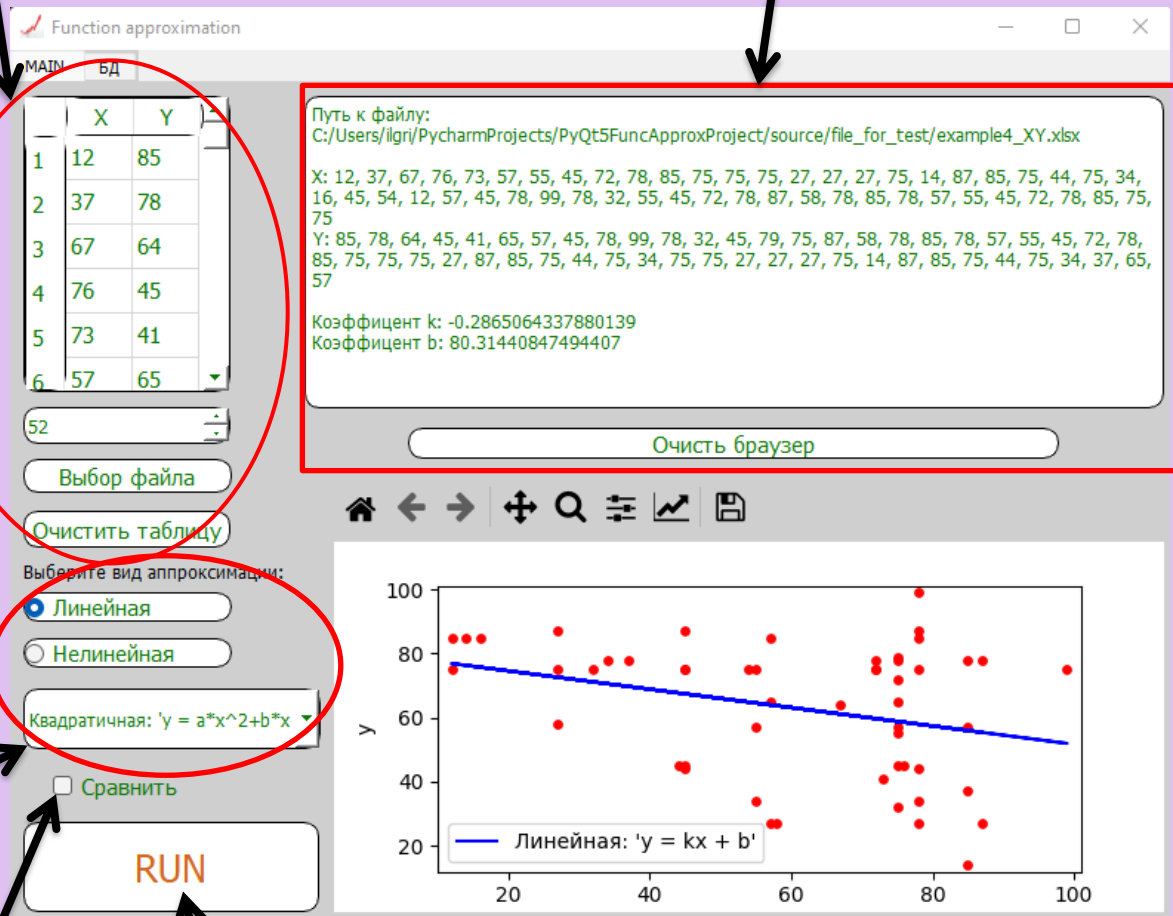
**Ввод точек**  
Можно вручную, или выбрать  
.XLSX файл в котором два  
столбца X и Y

**Текст браузер.**  
При выборе .XLSX файла сюда  
выводится его путь.  
При нажатие на кнопку **RUN**  
выводятся координаты точек.  
В случае линейной аппроксимации  
выводятся коэффициенты k и b.

**Выбор  
параметров  
аппроксимации**

**Возможность  
наложения одного  
графика на другой  
для сравнения**

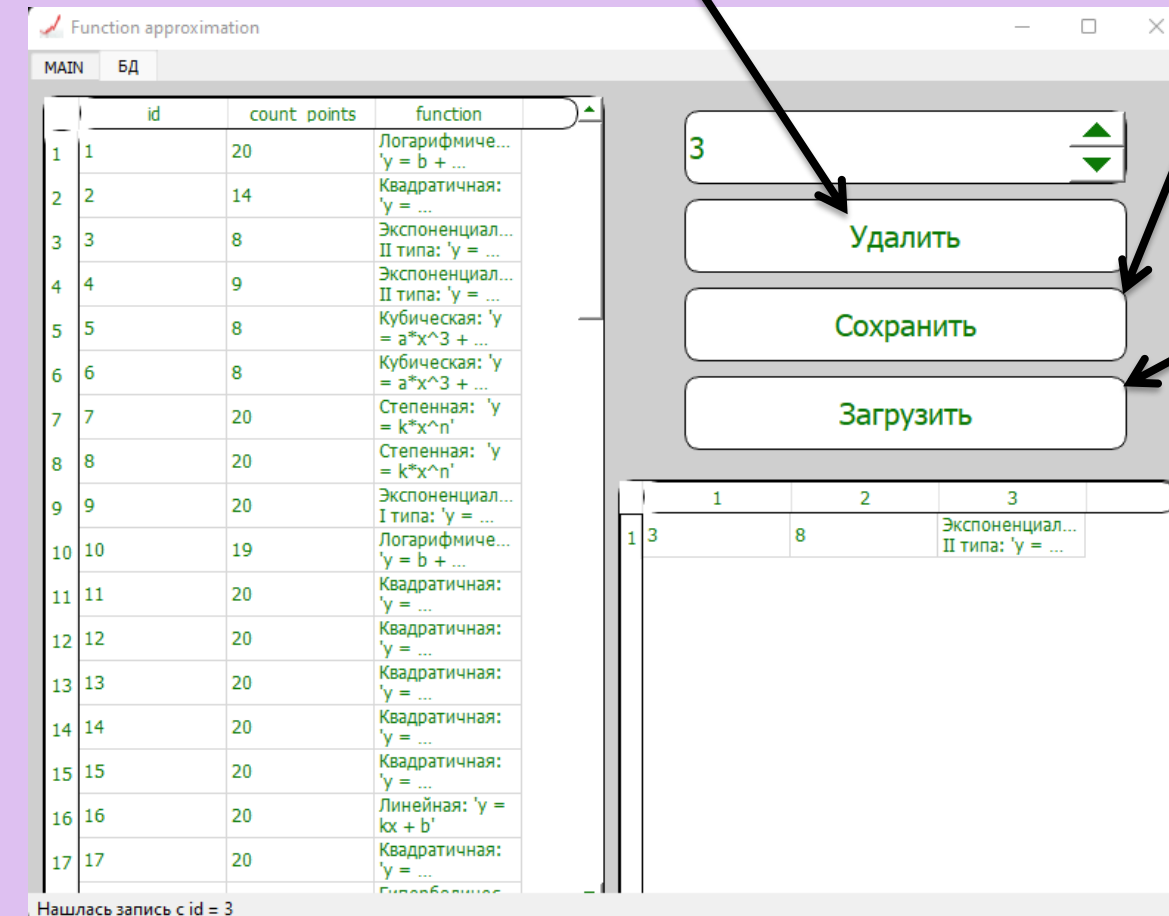
**Кнопка вывода графика и  
записи параметров в БД**

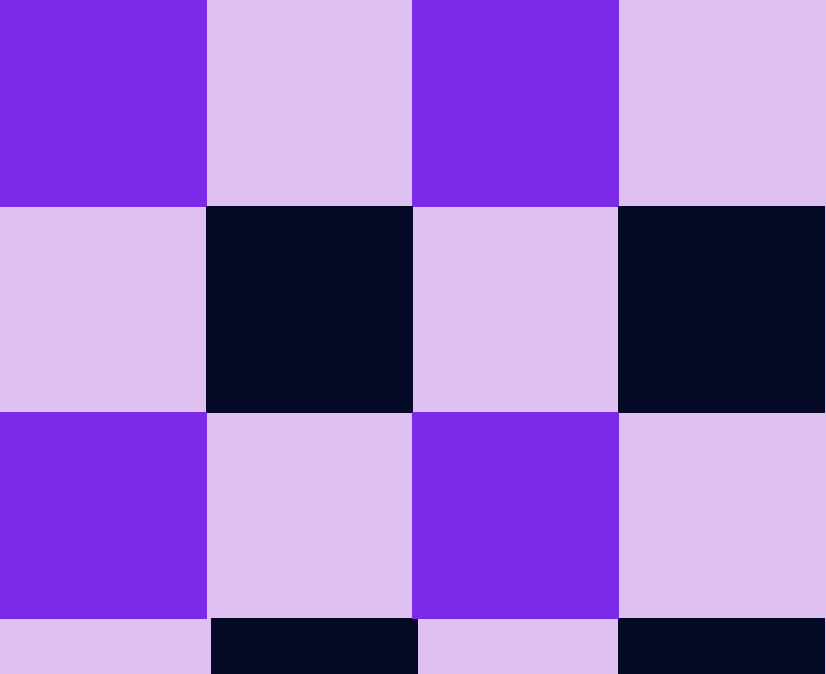


**Возможность удалить элемент  
по ID**

**Сохранить изменения в БД**

**Вывод ячейки из БД**





# Возможности для доработки

1

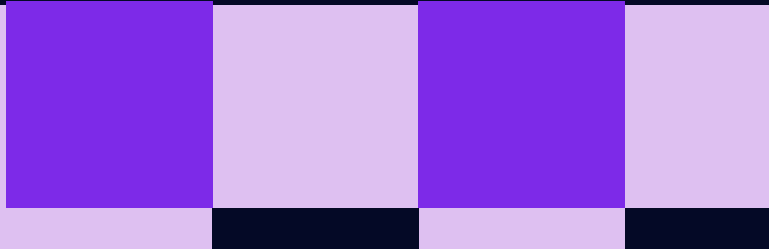
Добавить возможность интерполяции(способ нахождения промежуточного значений величины по набору известных значений)

2

Добавить возможность экстраполяции(способ определение значений функции за пределами первоначально известного интервала)

3

Добавить возможность по выбранному ID ячейки из базы данных, воспроизвести график.



# Спасибо за просмотр!

