



Министерство науки и высшего образования Российской Федерации  
Калужский филиал  
федерального государственного бюджетного  
образовательного учреждения высшего образования  
«Московский государственный технический университет имени Н.Э. Баумана  
(национальный исследовательский университет)»  
(КФ МГТУ им. Н.Э. Баумана)

**ФАКУЛЬТЕТ** ИУК «Информатика и управление»

**КАФЕДРА** ИУК4 «Программное обеспечение ЭВМ, информационные технологии»

## ДОМАШНЯЯ РАБОТА №1

**«Использование библиотеки PyQt для создания приложения с графическим интерфейсом»**

**ДИСЦИПЛИНА: «Перспективные языки программирования»**

Выполнил: студент гр. ИУК4-32Б

Зудин Д.В. (Подпись) (Зудин Д.В. (Ф.И.О.))

Проверил:

Пчелинцева Н.И. (Подпись) (Пчелинцева Н.И. (Ф.И.О.))

Дата сдачи (защиты):

24.22

Результаты сдачи (защиты):

- Балльная оценка:

100

- Оценка:

зачено

Калуга, 2022 г.

**Цель:** приобретение практических навыков разработки графического интерфейса с помощью библиотеки PyQt средствами языка Python.

**Задачи:**

1. Ознакомиться с конструкцией библиотеки PyQt;
2. Изучить способы создания мини-приложений с помощью PyQt;
3. Изучить возможности данной платформы;
4. Закрепить полученные в ходе выполнения домашней работы навыки.

## Вариант №20

### Формулировка задания №1

В созданном окне вам необходимо построить график функций на осях x и y по заданной формуле:

$$\cos^3 x \sin x - \frac{5x}{\sin 2x \cos x}$$

### Формулировка задания №2

Используя предыдущее окно с графиком, вам необходимо:

1. Задать отдельное окно для вашего графика (окно подразумевает отдельную рамку для графика в главном окне).
2. Создать поле ввода, позволяющее пополнить количество товара на складе на N единиц (число N вводится пользователем). Разделите окно на две части и разместите элемент пополнения и кнопку «Пополнить» в левой части окна. В правой части должно быть текстовое поле, принимающее от пользователя число, определяющее количество товара, которое им было куплено. При нажатии кнопки «Купить» должно выводиться сообщение об успешной покупке (если товара на складе достаточно) и из склада должен вычитаться купленный товар, или же сообщение о том, что на складе недостаточно товара и что его необходимо пополнить.

Пополнить	В наличии	Купить
<input type="text"/>	<input type="text"/>	<input type="text"/>

**Недостаточно товара на складе!**

### Формулировка задания №3

Создайте Button с названием «Справка», при наведении на которую всплывает подсказка, из которой можно узнать ФИО и номер группы студента, выполняющего домашнее задание. При нажатии на кнопку в окне ввода-вывода появляется сообщение «Обращение за справкой».

### Листинг интерфейса программы

```
# -*- coding: utf-8 -*-

# Form implementation generated from reading ui file 'Application.ui'
#
# Created by: PyQt5 UI code generator 5.15.4
#
# WARNING: Any manual changes made to this file will be lost when pyuic5 is
# run again. Do not edit this file unless you know what you are doing.

from PyQt5 import QtCore, QtGui, QtWidgets

class Ui_MainWindow(object):
    def setupUi(self, MainWindow):
        """Настройки инициализации"""
        MainWindow.setObjectName("MainWindow")
        MainWindow.resize(807, 617)

        #Главный виджет
        self.centralwidget = QtWidgets.QWidget(MainWindow)
        self.centralwidget.setObjectName("centralwidget")

        #TabWidget Главное меню
        self.MainMenu = QtWidgets.QTabWidget(self.centralwidget)
        self.MainMenu.setGeometry(QtCore.QRect(-10, -30, 821, 651))
        self.MainMenu.setToolTip("")
        self.MainMenu.setStyleSheet("background-color: #22222e")
        self.MainMenu.setTabShape(QtWidgets.QTabWidget.Rounded)
        self.MainMenu.setTabsClosable(True)
        self.MainMenu.setMovable(False)
        self.MainMenu.setObjectName("MainMenu")
```

```

#Страница меню
self.menu = QtWidgets.QWidget()
self.menu.setStyleSheet("background-color: #22222e")
self.menu.setObjectName("menu")

#Кнопка 1-ого задания
self.btn1 = QtWidgets.QPushButton(self.menu)
self.btn1.setGeometry(QtCore.QRect(230, 180, 380, 60))
font = QtGui.QFont()
font.setPointSize(14)
font.setBold(True)
font.setWeight(75)
self.btn1.setFont(font)
self.btn1.setStyleSheet("QPushButton {\n"
    "    background-color: #22222e;\n"
    "    border: 2px solid #f66867;\n"
    "    border-radius: 30;\n"
    "    color: white;\n"
    "}\n"
    "\n"
    "QPushButton: pressed {\n"
    "    background-color: #0d0d12\n"
    "}")
self.btn1.setObjectName("btn1")

#Кнопка 2-ого задания
self.btn2 = QtWidgets.QPushButton(self.menu)
self.btn2.setGeometry(QtCore.QRect(230, 270, 380, 60))
font = QtGui.QFont()
font.setPointSize(14)
font.setBold(True)
font.setWeight(75)
self.btn2.setFont(font)
self.btn2.setStyleSheet("QPushButton {\n"
    "    background-color: #22222e;\n"
    "    border: 2px solid #f66867;\n"
    "    border-radius: 30;\n"
    "    color: white;\n"
    "}\n"
    "\n"
    "QPushButton: pressed {\n"

```

```

        "    background-color: #0d0d12\n"
        "}")
self.btn2.setObjectName("btn2")

#Кнопка 3-его задания
self.btn3 = QtWidgets.QPushButton(self.menu)
self.btn3.setGeometry(QtCore.QRect(230, 360, 380, 60))
font = QtGui.QFont()
font.setPointSize(14)
font.setBold(True)
font.setWeight(75)
self.btn3.setFont(font)
self.btn3.setToolTip("Зудин Даниил Васильевич \nИУК4-32Б")
self.btn3.setStyleSheet("QPushButton {\n"
        "    background-color: #22222e;\n"
        "    border: 2px solid #f66867;\n"
        "    border-radius: 30;\n"
        "    color: white;\n"
        "}\n"
        "\n"
        "QPushButton: pressed {\n"
        "    background-color: #0d0d12\n"
        "}")
self.btn3.setObjectName("btn3")

#Кнопка выхода из приложения
self.btnexit1 = QtWidgets.QPushButton(self.menu)
self.btnexit1.setGeometry(QtCore.QRect(700, 580, 110, 30))
self.btnexit1.setStyleSheet("QPushButton {\n"
        "    background-color: #22222e;\n"
        "    border: 2px solid #f66867;\n"
        "    color: white;\n"
        "}\n"
        "\n"
        "QPushButton: pressed {\n"
        "    background-color: #0d0d12\n"
        "}")
self.btnexit1.setObjectName("btnexit1")

#Заглавный текст
self.lab1 = QtWidgets.QLabel(self.menu)
self.lab1.setGeometry(QtCore.QRect(230, 90, 380, 60))

```

```

font = QtGui.QFont()
font.setPointSize(18)
font.setBold(True)
font.setWeight(75)
self.lab1.setFont(font)
self.lab1.setStyleSheet("color: white;\n"
                        "background-color: #fb5b5d;")
self.lab1.setAlignment(QtCore.Qt.AlignCenter)
self.lab1.setObjectName("lab1")

#TabWidget Страница графика
self.MainMenu.addTab(self.menu, "")
self.graphic = QtWidgets.QWidget()
self.graphic.setObjectName("graphic")

#Кнопка выхода в главное меню
self.btnexit2 = QtWidgets.QPushButton(self.graphic)
self.btnexit2.setGeometry(QtCore.QRect(700, 580, 110, 30))
self.btnexit2.setStyleSheet("QPushButton {\n"
                            "    background-color: #22222e;\n"
                            "    background-color: #22222e;\n"
                            "    border: 2px solid #f66867;\n"
                            "    color: white;\n"
                            "}\n"
                            "\n"
                            "QPushButton: pressed {\n"
                            "    background-color: #0d0d12\n"
                            "}")
self.btnexit2.setObjectName("btnexit2")

self.widget = QtWidgets.QWidget(self.graphic)
self.widget.setGeometry(QtCore.QRect(10, 10, 810, 560))
self.widget.setObjectName("widget")

#TabWidget Страница склада
self.MainMenu.addTab(self.graphic, "")
self.warehouse = QtWidgets.QWidget()
self.warehouse.setObjectName("warehouse")

#Кнопка выхода в главное меню
self.btnexit3 = QtWidgets.QPushButton(self.warehouse)
self.btnexit3.setGeometry(QtCore.QRect(700, 580, 110, 30))

```

```

self.btnexit3.setStyleSheet("QPushButton {\n"
                             "    background-color: #22222e;\n"
                             "    border: 2px solid #f66867;\n"
                             "    color: white;\n"
                             "}\n"
                             "\n"
                             "QPushButton: pressed {\n"
                             "    background-color: #0d0d12\n"
                             "}")

self.btnexit3.setObjectName("btnexit3")

#Ввод текста "ПОПОЛНИТЬ"
self.le1 = QtWidgets.QLineEdit(self.warehouse)
self.le1.setGeometry(QtCore.QRect(120, 310, 190, 40))
font = QtGui.QFont()
font.setPointSize(12)
font.setBold(True)
font.setWeight(75)
self.le1.setFont(font)
self.le1.setStyleSheet("color: white")
self.le1.setText("")
self.le1.setAlignment(QtCore.Qt.AlignCenter)
self.le1.setObjectName("le1")

#Ввод текста "КУПИТЬ"
self.le2 = QtWidgets.QLineEdit(self.warehouse)
self.le2.setGeometry(QtCore.QRect(540, 310, 190, 40))
font = QtGui.QFont()
font.setPointSize(12)
font.setBold(True)
font.setWeight(75)
self.le2.setFont(font)
self.le2.setStyleSheet("color: white")
self.le2.setText("")
self.le2.setAlignment(QtCore.Qt.AlignCenter)
self.le2.setObjectName("le2")

#Вывод количества товара на складе
self.lab2 = QtWidgets.QLabel(self.warehouse)
self.lab2.setGeometry(QtCore.QRect(330, 310, 190, 40))
font = QtGui.QFont()
font.setPointSize(12)

```

```

font.setBold(True)
font.setWeight(75)
self.lab2.setFont(font)
self.lab2.setStyleSheet("color: white;\n"
                        "background-color: #fb5b5d;")

self.lab2.setText("")
self.lab2.setAlignment(QtCore.Qt.AlignCenter)
self.lab2.setObjectName("lab2")

#Текст "В наличии"
self.lab3 = QtWidgets.QLabel(self.warehouse)
self.lab3.setGeometry(QtCore.QRect(330, 250, 190, 40))
font = QtGui.QFont()
font.setPointSize(12)
font.setBold(True)
font.setWeight(75)
self.lab3.setFont(font)
self.lab3.setStyleSheet("color: white;\n"
                        "background-color: #fb5b5d;")
self.lab3.setAlignment(QtCore.Qt.AlignCenter)
self.lab3.setObjectName("lab3")

#Кнопка "Пополнить"
self.btn4 = QtWidgets.QPushButton(self.warehouse)
self.btn4.setGeometry(QtCore.QRect(120, 250, 190, 40))
font = QtGui.QFont()
font.setPointSize(12)
font.setBold(True)
font.setWeight(75)
self.btn4.setFont(font)
self.btn4.setStyleSheet("QPushButton {\n"
                        "    background-color: #22222e;\n"
                        "    border: 2px solid #f66867;\n"
                        "    border-radius: 30;\n"
                        "    color: white;\n"
                        "}\n"
                        "\n"
                        "QPushButton: pressed {\n"
                        "    background-color: #0d0d12\n"
                        "}")
self.btn4.setObjectName("btn4")

```



```

#Кнопка "Купить"
self.btn5 = QtWidgets.QPushButton(self.warehouse)
self.btn5.setGeometry(QtCore.QRect(540, 250, 190, 40))
font = QtGui.QFont()
font.setPointSize(12)
font.setBold(True)
font.setWeight(75)
self.btn5.setFont(font)
self.btn5.setStyleSheet("QPushButton {\n"
                        "    background-color: #22222e;\n"
                        "    border: 2px solid #f66867;\n"
                        "    border-radius: 30;\n"
                        "    color: white;\n"
                        "}\n"
                        "\n"
                        "QPushButton: pressed {\n"
                        "    background-color: #0d0d12\n"
                        "}")

self.btn5.setObjectName("btn5")

#Текст после совершения операции
self.lab4 = QtWidgets.QLabel(self.warehouse)
self.lab4.setGeometry(QtCore.QRect(120, 370, 611, 40))
font = QtGui.QFont()
font.setPointSize(12)
font.setBold(True)
font.setWeight(75)
self.lab4.setFont(font)
self.lab4.setStyleSheet("color: white;\n"
                        "background-color: #fb5b5d;")
self.lab4.setInputMethodHints(QtCore.Qt.ImhNone)
self.lab4.setAlignment(QtCore.Qt.AlignCenter)
self.lab4.setObjectName("lab4")
self.lab4.setHidden(True)

self.MainMenu.addTab(self.warehouse, "")
MainWindow.setCentralWidget(self.centralwidget)

self.retranslateUi(MainWindow)
self.MainMenu.setCurrentIndex(0)
QtCore.QMetaObject.connectSlotsByName(MainWindow)

```

```

def retranslateUi(self, MainWindow):
    """Текст в элементах"""
    _translate = QtCore.QCoreApplication.translate
    MainWindow.setWindowTitle(_translate("MainWindow", "Домашняя работа
№1"))

    self.btn1.setText(_translate("MainWindow", "Задание №1 (График)"))
    self.btn2.setText(_translate("MainWindow", "Задание №2 (Склад)"))
    self.btn3.setText(_translate("MainWindow", "Задание №3 (Справка)"))
    self.btnexit1.setText(_translate("MainWindow", "Выход"))
    self.lab1.setText(_translate("MainWindow", "Основы PyQt"))
    self.MainMenu.setTabText(self.MainMenu.indexOf(self.menu),
_translate("MainWindow", "menu"))
    self.btnexit2.setText(_translate("MainWindow", "Назад"))
    self.MainMenu.setTabText(self.MainMenu.indexOf(self.graphic),
_translate("MainWindow", "graphic"))
    self.btnexit3.setText(_translate("MainWindow", "Назад"))
    self.lab3.setText(_translate("MainWindow", "В наличии"))
    self.btn4.setText(_translate("MainWindow", "Пополнить"))
    self.btn5.setText(_translate("MainWindow", "Купить"))
    self.lab4.setText(_translate("MainWindow", "Недостаточно товара на
складе!"))

    self.MainMenu.setTabText(self.MainMenu.indexOf(self.warehouse),
_translate("MainWindow", "warehouse"))

```

### **Листинг программы для задания**

```

import sys
from PyQt5 import QtWidgets
from ui import Ui_MainWindow

from matplotlib.backends.backend_qt5agg import FigureCanvasQTAgg
from matplotlib.figure import Figure
import numpy as np

class Homework(QtWidgets.QMainWindow):
    """Основной класс для логики"""
    quantity = 0

    def __init__(self):
        """Инициализация наследуемого класса"""
        super(Homework, self).__init__()
        self.ui = Ui_MainWindow()

```

```

self.ui.setupUi(self)
self.init_UI()

def init_UI(self):
    """Основная логика программы"""
    self.ui.btn1.clicked.connect(self.task_1)
    self.ui.btn2.clicked.connect(self.task_2)
    self.ui.btn3.clicked.connect(lambda: print("Обращение за справкой"))
    self.ui.btn4.clicked.connect(self.fill)
    self.ui.btn5.clicked.connect(self.buy)
    self.ui.btnexit1.clicked.connect(sys.exit)
    self.ui.btnexit2.clicked.connect(lambda:
self.ui.MainMenu.setCurrentIndex(0))
    self.ui.btnexit3.clicked.connect(lambda:
self.ui.MainMenu.setCurrentIndex(0))

def task_1(self):
    """Первое задание"""
    self.ui.MainMenu.setCurrentIndex(1)
    fig = Figure(figsize=(5, 5))
    can = FigureCanvasQTAgg(fig)
    layout = QtWidgets.QVBoxLayout(self.ui.widget)
    layout.addWidget(can)
    ax = can.figure.add_subplot(111)
    ax.cla()

    x = np.arange(-10, 10, 0.1)
    y = np.cos(x) * np.cos(x) * np.cos(x) * np.sin(x)
    for i in range(0, len(y)):
        y[i] = y[i] - (5 * x[i]) / (np.sin(2 * x[i]) * np.cos(x[i]))

    ax.plot(x, y)
    ax.grid(True)
    can.figure.tight_layout()
    can.draw()

def task_2(self):
    """Второе задание"""
    self.ui.MainMenu.setCurrentIndex(2)
    self.ui.lab4.setHidden(True)

def fill(self):

```

```

        """Пополнение склада"""
        self.ui.lab2.clear()
        fill_up = self.ui.le1.text()
        self.ui.le1.clear()
        self.quantity += int(fill_up)
        self.ui.lab2.setText(str(self.quantity))
        self.ui.lab4.setText("Пополнение выполнено успешно!")
        self.ui.lab4.setHidden(False)

def buy(self):
    """Покупка"""
    self.ui.lab2.clear()
    fill_down = self.ui.le2.text()
    if self.quantity < int(fill_down):
        self.ui.lab4.setText("Недостаточно товара на складе!")
    else:
        self.quantity -= int(fill_down)
        self.ui.lab4.setText("Покупка выполнена успешно!")
    self.ui.le2.clear()
    self.ui.lab2.setText(str(self.quantity))
    self.ui.lab4.setHidden(False)

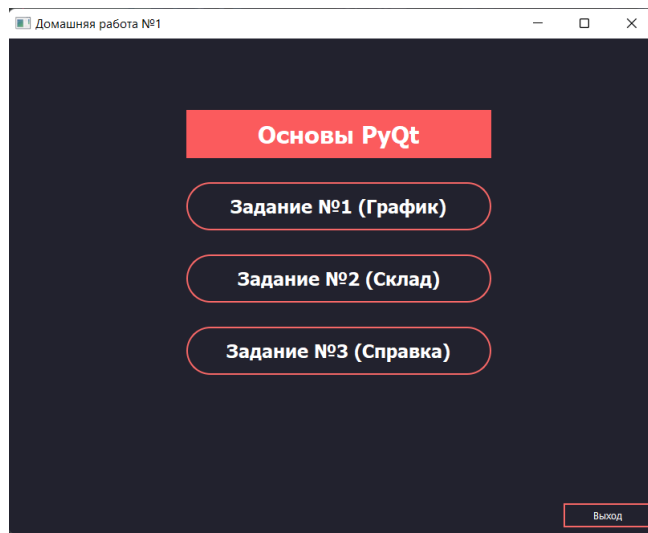
app = QtWidgets.QApplication([])
application = Homework()
application.show()

sys.exit(app.exec())

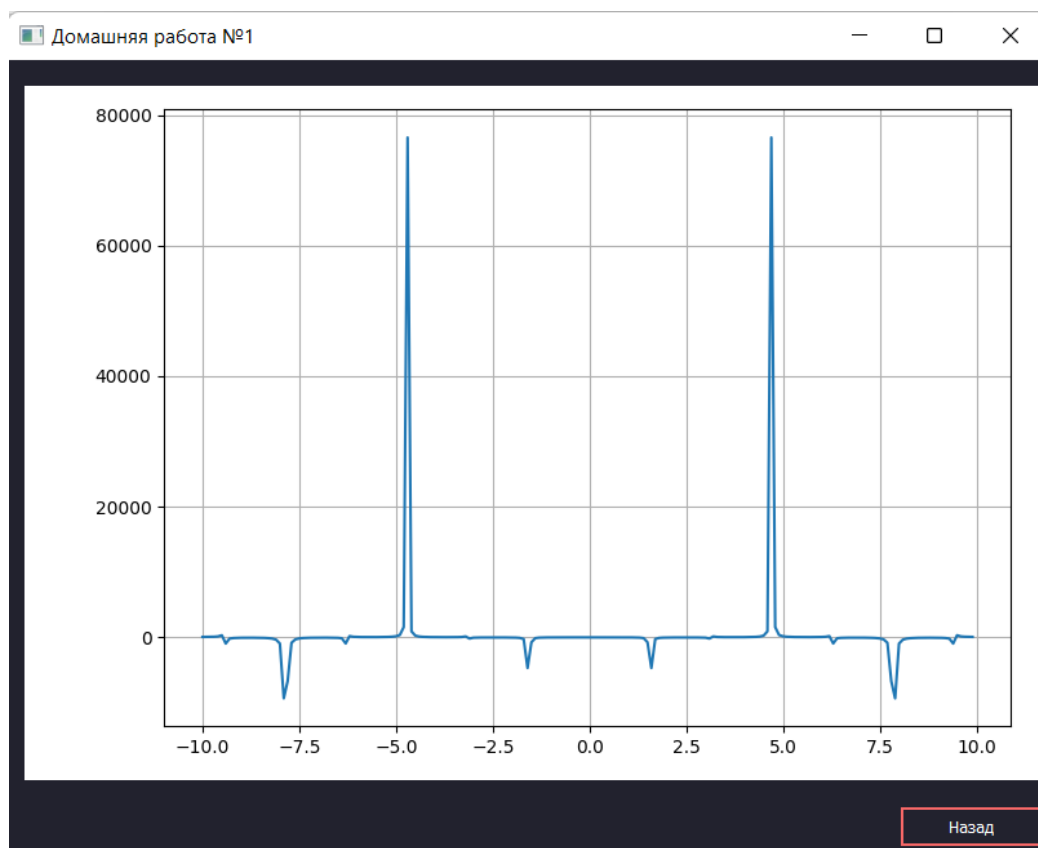
```

## **Результат выполнения программы для задания**

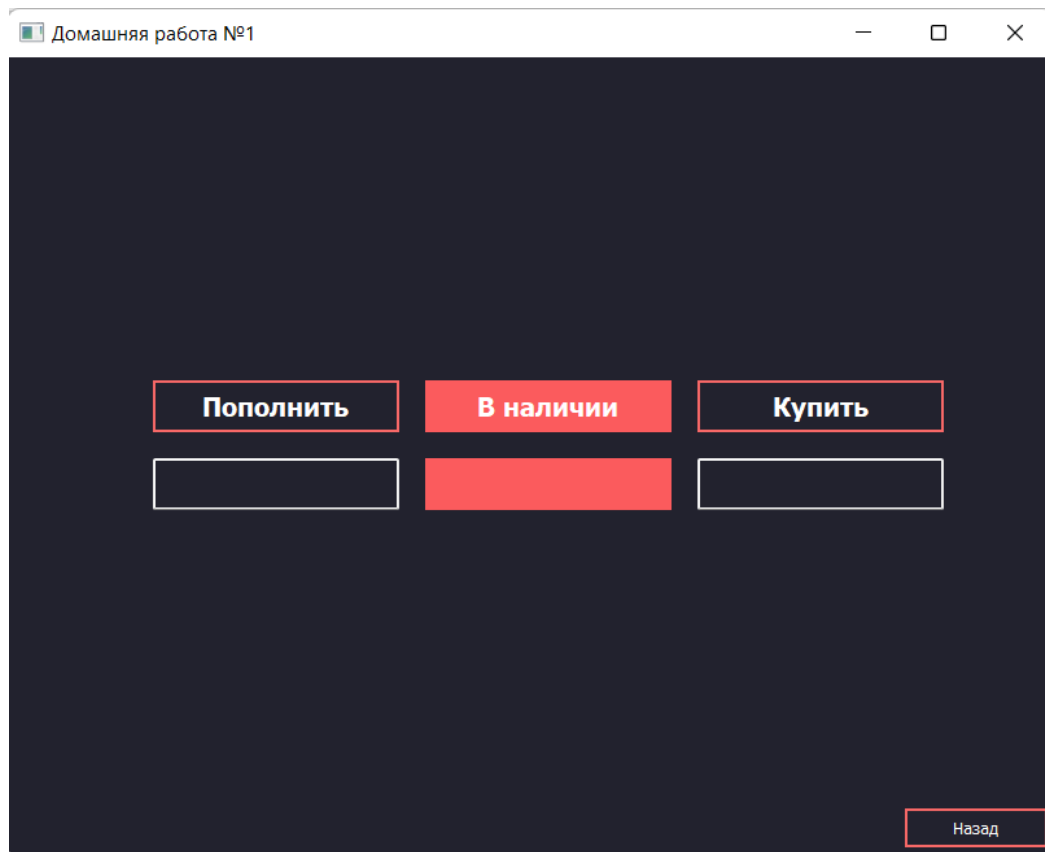
### **Меню**



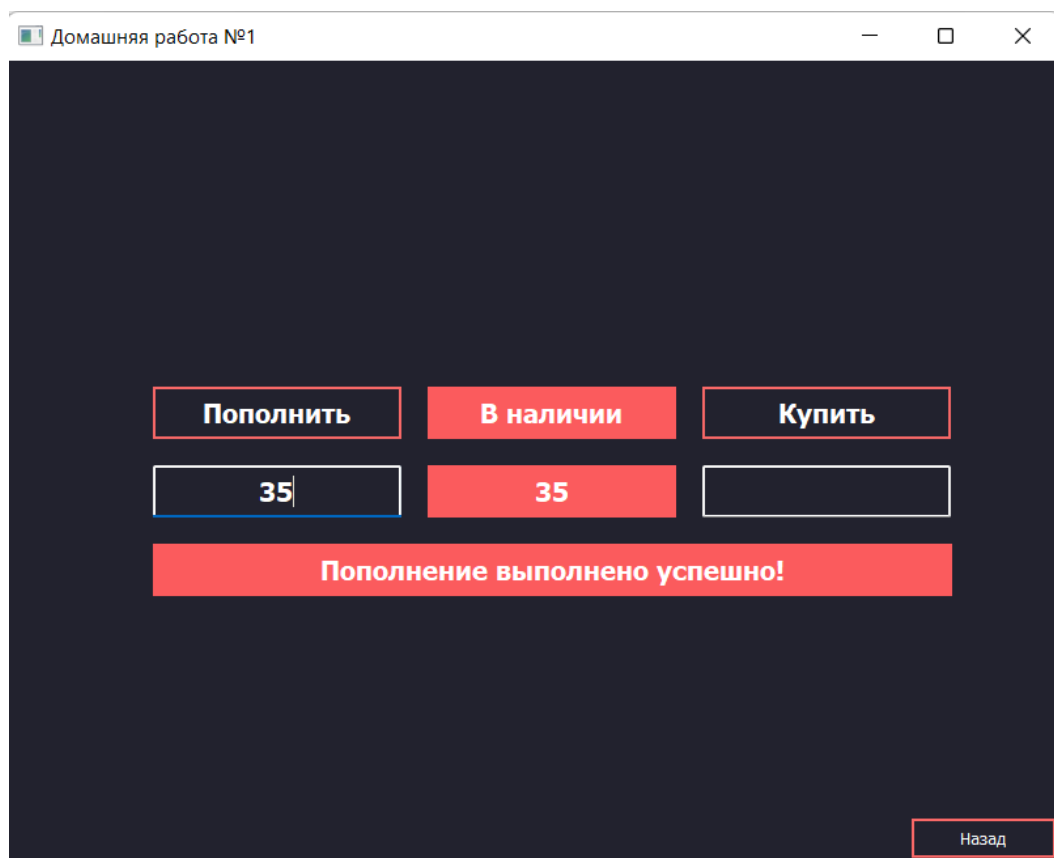
### Задание №1 (График)



### Задание №2 (Склад)



### Пополнение склада



### Успешная покупка

Домашняя работа №1

Пополнить

В наличии

Купить

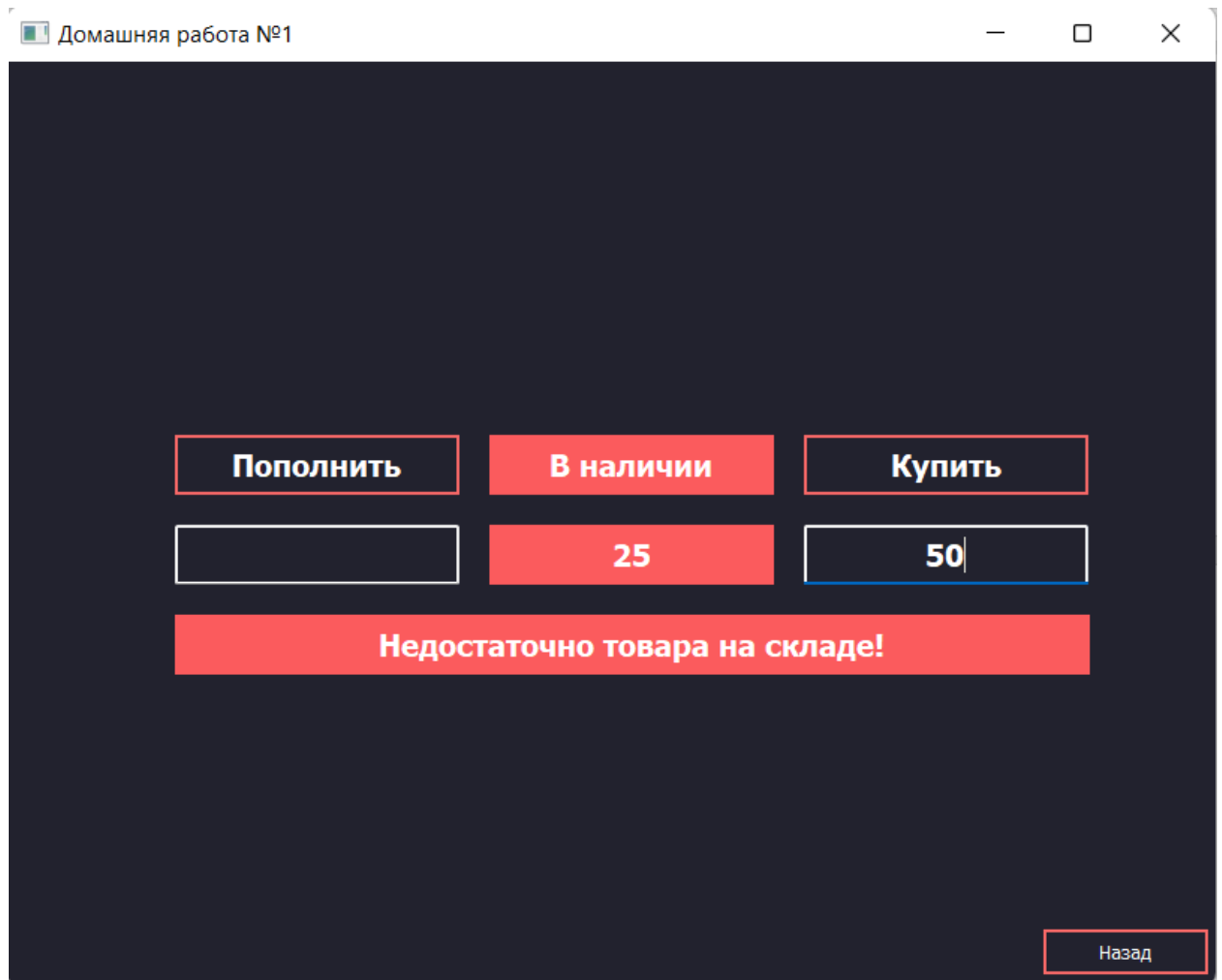
25

10

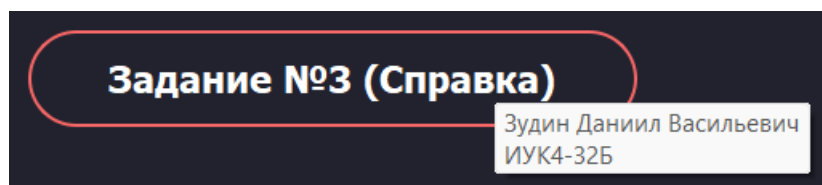
Покупка выполнена успешно!

Назад

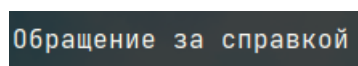
Неуспешная покупка



### Задание №3 (Справка)



### Окно ввода-вывода



### Выводы:

В ходе работы были приобретены практические навыки разработки графического интерфейса с помощью библиотеки PyQt средствами языка Python.