



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 ИУК2 "Информационные системы и сети"

ЛАБОРАТОРНАЯ РАБОТА №4

«ДОПОЛНИТЕЛЬНЫЕ БИБЛИОТЕКИ ЯЗЫКА PYTHON »

ДИСЦИПЛИНА: «Перспективные языки программирования»

Выполнил: студент гр. ИУК4-33Б _____ (____ Сароян Н.А.____)
(Подпись) (Ф.И.О.)

Проверил: _____ (____ Осипова О.В.____)
(Подпись) (Ф.И.О.)

Дата сдачи (защиты):

Результаты сдачи (защиты):

- Балльная оценка:

- Оценка:

Калуга, 2024

ЦЕЛИ: Целью выполнения домашней работы является приобретение практических навыков разработки графического интерфейса с помощью библиотеки PyQt средствами языка Python.

ЗАДАЧИ: Основными задачами выполнения домашней работы являются:

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 5

Задача 1. В созданном окне вам необходимо построить график функций на осях x и y по заданной формуле:

$$\frac{|\sin(5x)|}{\sin(x - 1)}$$

Задача 2. Используйте предыдущее окно с нарисованным графиком, вам необходимо:

1. Задать нумерацию оси координат вашего графика
2. Создайте список продуктов и `CheckBox` выбора для каждого из них. Напишите цену каждого из продуктов после названия. При выборе всех нужных пользователю продуктов и нажатии на кнопку должно выводиться сообщение об

итоговой цене покупки(Рис.35)

✓	товар1, 10	Цена покупки 96
✓	товар2, 50	
✓	товар3, 4	
	товар4,15	
✓	товар5,98	
	товар6,42	
	товар7,12	

Рис.35.Пример реализации задания

3. Организовать Message Box, при нажатии на которую выводится сообщение, содержащее:

- ФИО;
- Специальность;
- Группа.

Листинг программы 1

```
import sys

import numpy as np

import matplotlib.pyplot as plt

from matplotlib.backends.backend_qt5agg import FigureCanvasQTAgg as FigureCanvas

from PyQt5.QtWidgets import QApplication, QMainWindow, QVBoxLayout, QWidget

class PlotWindow(QMainWindow):

    def __init__(self):
```

```
super().__init__()

self.setWindowTitle("График функции")

self.setGeometry(100, 100, 800, 600)

self.main_widget = QWidget(self)
self.setCentralWidget(self.main_widget)
layout = QVBoxLayout(self.main_widget)

self.canvas = FigureCanvas(plt.figure())
layout.addWidget(self.canvas)

self.plot_graph()

def plot_graph(self):

    x = np.linspace(1.1, 10, 1000)
    y = np.abs(np.sin(5 * x)) / np.sin(x - 1)

    ax = self.canvas.figure.add_subplot(111)
    ax.plot(x, y)
    ax.set_title("График функции  $|\sin(5x)| / \sin(x - 1)$ ")
    ax.set_xlabel("x")
    ax.set_ylabel("y")
    ax.grid(True)

    self.canvas.draw()

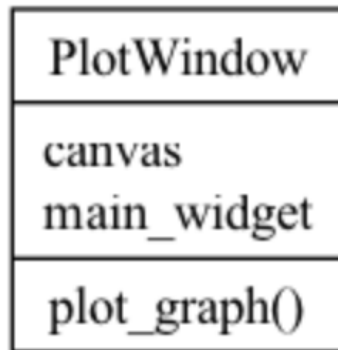
if __name__ == "__main__":
    app = QApplication(sys.argv)
```

```
main_window = PlotWindow()

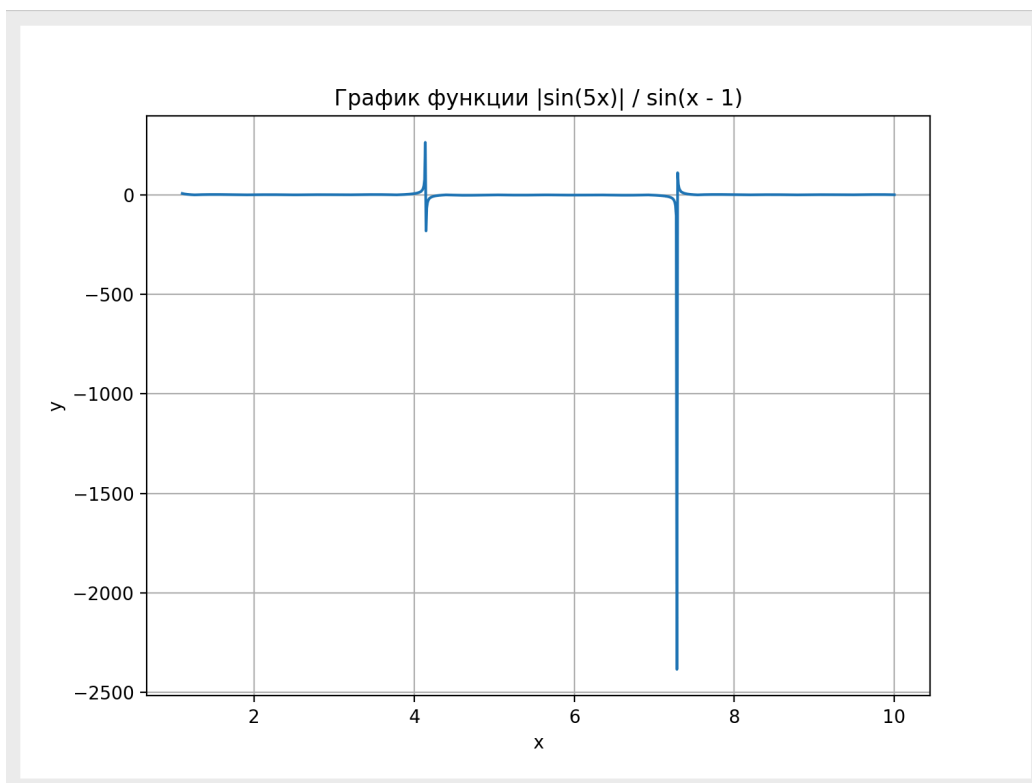
main_window.show()

sys.exit(app.exec_())
```

UML диаграмма:



Результат выполнения программы:



Листинг программы 2

```
import sys

from PyQt5.QtWidgets import QVBoxLayout, QHBoxLayout, QCheckBox, QPushButton,
QLabel, QMainWindow, QWidget, QApplication

class ProductWindow(QMainWindow):

    def __init__(self):
        super().__init__()

        self.setWindowTitle("Список продуктов")

        self.setGeometry(100, 100, 400, 400)

        self.main_widget = QWidget(self)

        self.setCentralWidget(self.main_widget)

        layout = QVBoxLayout(self.main_widget)

        self.products = [

            ("Товар1", 10),

            ("Товар2", 50),

            ("Товар3", 4),

            ("Товар4", 15),

            ("Товар5", 98),

            ("Товар6", 42),

            ("Товар7", 12)

        ]

        self.checkboxes = []
```

```
        for product, price in self.products:

            checkbox = QCheckBox(f"{product}, {price}")

            layout.addWidget(checkbox)

            self.checkboxes.append((checkbox, price))

        self.calculate_button = QPushButton("Цена покупки")

        self.calculate_button.clicked.connect(self.calculate_total_price)

        layout.addWidget(self.calculate_button)

        self.total_label = QLabel("Итоговая цена: 0")

        layout.addWidget(self.total_label)

    def calculate_total_price(self):

        total_price = 0

        for checkbox, price in self.checkboxes:

            if checkbox.isChecked():

                total_price += price

        self.total_label.setText(f"Итоговая цена: {total_price}")

if __name__ == "__main__":

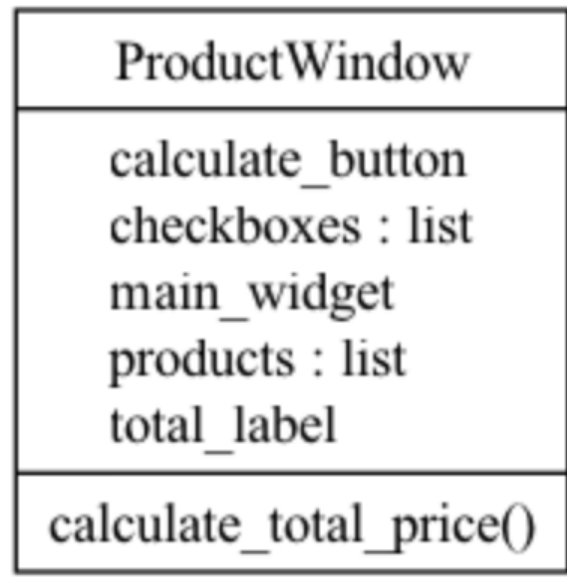
    app = QApplication(sys.argv)

    product_window = ProductWindow()

    product_window.show()

    sys.exit(app.exec_())
```

UML диаграмма:



Результат выполнения программы:

The screenshot shows a shopping application interface. It features a list of seven items, each with a checkbox and a label: "товар1, 10", "товар2, 50", "товар3, 4", "товар4, 15", "товар5, 98", "товар6, 42", and "товар7, 12". The checkboxes for "товар3, 4", "товар4, 15", and "товар5, 98" are checked. Below the list is a text input field labeled "Цена покупки". At the bottom left, the text "Итоговая цена: 117" is displayed.

Листинг программы 3

```
import sys

from PyQt5.QtWidgets import QMessageBox, QPushButton, QMainWindow, QApplication

class InfoWindow(QMainWindow):

    def __init__(self):
        super().__init__()

        self.setWindowTitle("Информация о пользователе")

        self.setGeometry(100, 100, 300, 200)

        self.info_button = QPushButton("Показать информацию", self)

        self.info_button.clicked.connect(self.show_info_message)

        self.setCentralWidget(self.info_button)

    def show_info_message(self):

        fio = "Сароян Норик Аветикович"

        specialty = "Программная инженерия"

        group = "ИУК4-33Б"

        message = QMessageBox(self)

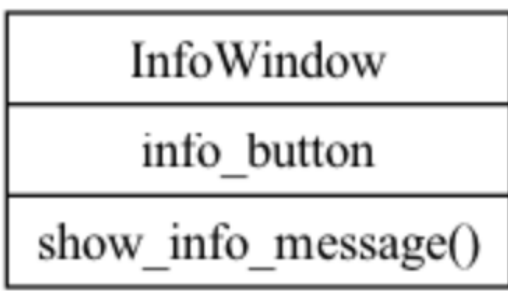
        message.setWindowTitle("Информация")

        message.setText(f"ФИО: {fio}\nСпециальность: {specialty}\nГруппа: {group}")

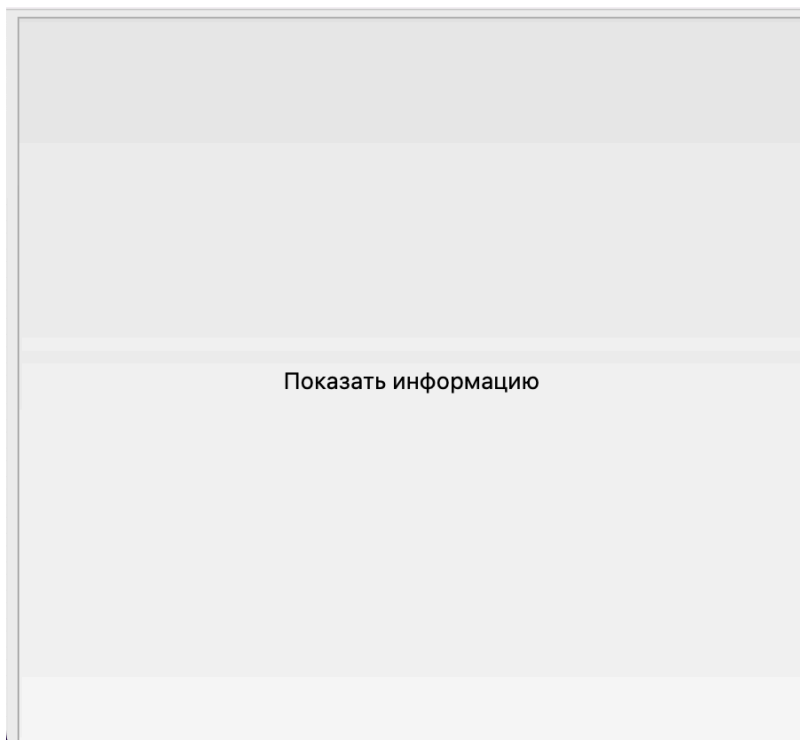
        message.exec_()
```

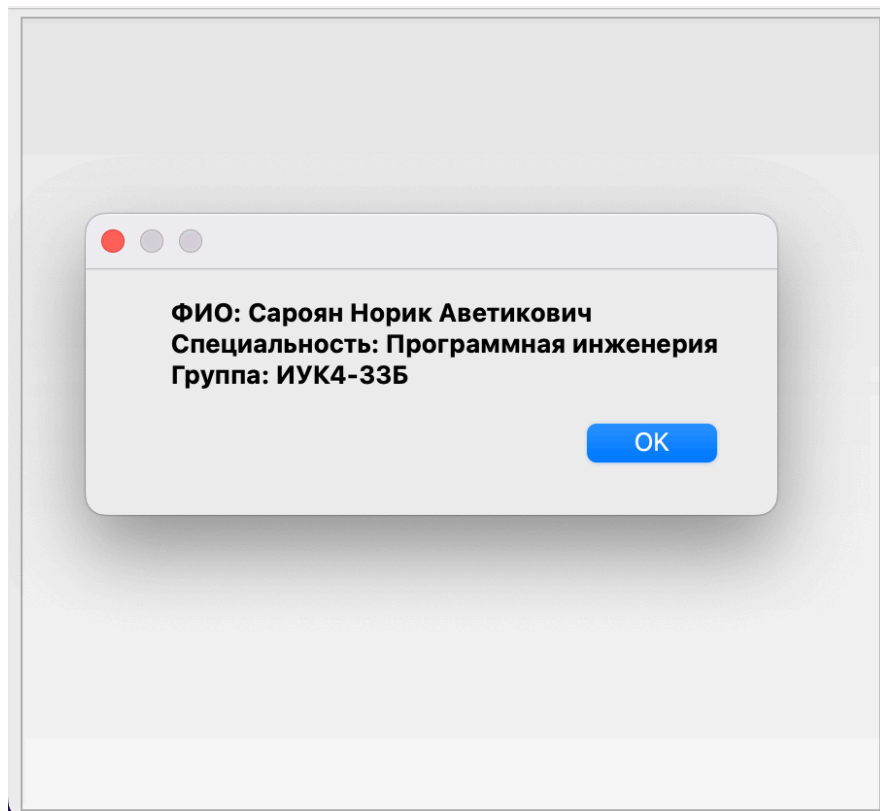
```
if __name__ == "__main__":  
    app = QApplication(sys.argv)  
    info_window = InfoWindow()  
    info_window.show()  
    sys.exit(app.exec_())
```

UML Диаграмма:



Результат выполнения программы:





Вывод: в ходе выполнения работы были приобретены практические навыки необходимые для разработки задач, решение которых предполагает использование сторонней библиотеки PyQt.