

Министерство науки и высшего образования Российской Федерации
Федеральное государственное автономное образовательное учреждение
высшего образования
«СЕВЕРО-КАВКАЗСКИЙ ФЕДЕРАЛЬНЫЙ УНИВЕРСИТЕТ»

Институт перспективной инженерии
Департамент цифровых, робототехнических систем и электроники

ОТЧЕТ
ПО ЛАБОРАТОРНОЙ РАБОТЕ №7
дисциплины
«Объектно-ориентированное программирование»
Вариант 13

Выполнил:
Мотовилов Вадим Борисович
3 курс, группа ИВТ-б-о-22-1,
09.03.01 «Информатика и
вычислительная техника»,
направленность (профиль)
«Программное обеспечение средств
вычислительной техники и
автоматизированных систем», очная
форма обучения

(подпись)

Проверил:
Воронкин Р.А.

(подпись)

Отчет защищен с оценкой _____ Дата защиты _____

Ставрополь, 2024 г.

Тема: Основы работы с Tkinter

Цель: приобретение навыков построения графического интерфейса пользователя GUI с помощью пакета Tkinter языка программирования Python версии 3.x.

Порядок выполнения работы:

Ссылка на гит: https://github.com/AkselSukub/OOP_7

1. Выполнил индивидуальное задание №1: напишите простейший калькулятор, состоящий из двух текстовых полей, куда пользователь вводит числа, и четырех кнопок "+", "-", "*", "/". Результат вычисления должен отображаться в метке. Если арифметическое действие выполнить невозможно (например, если были введены буквы, а не числа), то в метке должно появляться слово «ошибка».

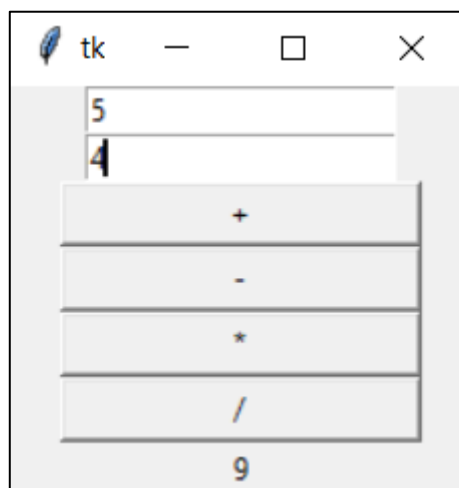


Рисунок 1. Результат выполнения программы

2. Выполнил индивидуальное задание №2: напишите программу, состоящую из семи кнопок, цвета которых соответствуют цветам радуги. При нажатии на ту или иную кнопку в текстовое поле должен вставляться код цвета, а в метку – название цвета.



Рисунок 2. Результат выполнения программы

3. Выполнил индивидуальное задание №3: перепишите программу из пункта 8 так, чтобы интерфейс выглядел другим образом, а именно чтобы кнопки располагались горизонтально:



Рисунок 3. Результат выполнения программы

4. Выполнил индивидуальное задание №4: напишите программу, состоящую из однострочного и многострочного текстовых полей и двух кнопок «Открыть» и «Сохранить». При клике на первую должен открываться на чтение файл, чье имя указано в поле класса Entry, а содержимое файла должно загружаться в поле типа Text. При клике на вторую кнопку текст, введенный пользователем в экземпляр Text, должен сохраняться в файле под именем, которое пользователь указал в однострочном текстовом поле. Файлы будут читаться и записываться в том же каталоге, что и файл скрипта.

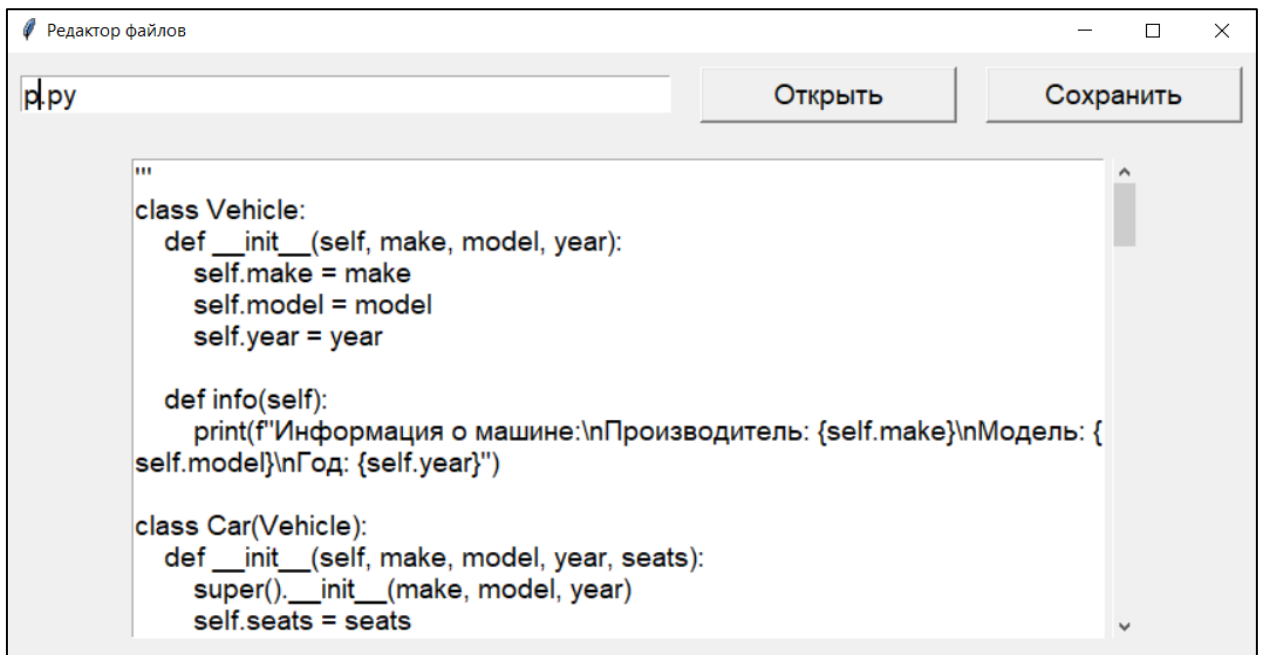


Рисунок 4. Результат выполнения программы

5. Выполнил индивидуальное задание №5: виджеты Radiobutton и Checkbutton поддерживают большинство свойств оформления внешнего вида, которые есть у других элементов графического интерфейса. При этом у Radiobutton есть особое свойство indicatoron. По-умолчанию он равен единице, в этом случае радиокнопка выглядит как нормальная радиокнопка. Однако если присвоить этой опции ноль, то виджет Radiobutton становится похожим на обычную кнопку по внешнему виду. Но не по смыслу.

Код:

```
#!/usr/bin/env python3
# -*- coding: utf-8 -*-

import tkinter as tk
from functools import partial

def calc(value: int, lable: tk.Label) -> None:
    match value:
        case 1:
            txt = "Первый radiobutton"
        case 2:
            txt = "Второй radiobutton"
        case 3:
            txt = "Третий radiobutton"
    lable.config(text=txt)

if __name__ == "__main__":
    root = tk.Tk()
```

```

fr = tk.Frame(root)
fr.pack(side="left")
var = tk.IntVar()
var.set(1)
rd1 = tk.Radiobutton(
    fr, text="1", value=1, width=10, height=5, indicatoron=False, variable=var
)
rd1.pack()
rd2 = tk.Radiobutton(
    fr, text="2", value=2, width=10, height=5, indicatoron=False, variable=var
)
rd2.pack()
rd3 = tk.Radiobutton(
    fr, text="3", value=3, width=10, height=5, indicatoron=False, variable=var
)
rd3.pack()
lable = tk.Label(root, text="Первый radiobutton")
lable.pack(side="left", anchor="center", fill="y", padx=30)

for rd in [rd1, rd2, rd3]:
    rd.config(command=partial(calc, rd["value"], lable))

root.mainloop()

```

Ответы на контрольные вопросы:

1. Какие существуют средства в стандартной библиотеке Python для построения графического интерфейса пользователя?

В стандартной библиотеке Python для построения графического интерфейса пользователя (GUI) существует несколько инструментов:

- Tkinter: Наиболее популярный и основной инструмент для создания GUI.
- Pygame: Используется для разработки игр, но также подходит для создания графического интерфейса.
- curses: Для создания текстовых интерфейсов в терминале (не графический, но часто используется для CLI).
- IDLE: Это встроенная среда разработки, использующая Tkinter для создания интерфейса.
- Turtle: Используется для обучения программированию через рисование и графику.

2. Что такое Tkinter?

Tkinter – это стандартная библиотека для создания графических интерфейсов пользователя в Python. Tkinter является оберткой вокруг библиотеки Tk, которая представляет собой набор инструментов для создания оконных приложений. Tkinter предоставляет доступ к множеству виджетов (кнопки, метки, текстовые поля и т.д.), которые можно использовать для создания GUI.

3. Какие требуется выполнить шаги для построения графического интерфейса с помощью Tkinter?

Основные шаги для создания графического интерфейса с использованием Tkinter:

- Импортировать Tkinter.
- Создать основное окно приложения (`root = tk.Tk()`).
- Добавить виджеты (например, кнопки, метки, текстовые поля) в окно.
- Разместить виджеты в окне (с помощью методов `pack()`, `grid()`, `place()`).
- Запустить главный цикл обработки событий (`root.mainloop()`).

4. Что такое цикл обработки событий?

Цикл обработки событий (или главный цикл) – это механизм, который позволяет программе реагировать на действия пользователя, такие как нажатия кнопок, движение мыши, изменения в полях ввода и т.д. Цикл обработки событий постоянно ожидает событий и вызывает соответствующие обработчики (функции) для этих событий.

Главный цикл запускается методом `root.mainloop()` и продолжается до тех пор, пока окно не будет закрыто.

5. Каково назначение экземпляра класса Tk при построении графического интерфейса с помощью Tkinter?

Экземпляр класса Tk является главным окном вашего приложения. Это окно, которое будет содержать все виджеты, с которыми взаимодействует пользователь. Он инициализирует внутренние механизмы Tkinter, включая

главный цикл событий, и управляет отображением всех элементов интерфейса.

6. Для чего предназначены виджеты Button, Label, Entry и Text?

Button: Кнопка, которая реагирует на действия пользователя (например, клики). С помощью неё можно вызвать функцию или выполнить команду.

Label: Метка (ярлык), отображающая текст или изображение, используется для вывода информации.

Entry: Однострочное текстовое поле, в котором пользователь может ввести данные.

Text: Многострочное текстовое поле, в котором можно редактировать большой объем текста (с возможностью прокрутки).

7. Каково назначение метода pack() при построении графического интерфейса пользователя?

Метод pack() используется для размещения виджетов в окне. Он автоматически размещает виджеты в контейнере (например, в окне или фрейме) по определенным правилам. Виджеты могут быть размещены вертикально или горизонтально, в зависимости от параметров.

8. Как осуществляется управление размещением виджетов с помощью метода pack()?

Метод pack() имеет несколько параметров для управления размещением виджетов:

- side: Определяет сторону контейнера (например, TOP, BOTTOM, LEFT, RIGHT).
- fill: Устанавливает, как виджет должен растягиваться по оси (например, X, Y, BOTH).
- expand: Если установлено в True, виджет будет расширяться, чтобы занять доступное пространство.

9. Как осуществляется управление полосами прокрутки в виджете Text?

Для управления полосами прокрутки в виджете Text используется

виджет `Scrollbar`. Полоса прокрутки связывается с виджетом `Text` через параметр `yscrollcommand`.

10. Для чего нужны тэги при работе с виджетом `Text`?

Тэги в `Text` используются для выделения или изменения атрибутов текста (например, цвет, стиль, шрифт). С помощью тэгов можно управлять стилями текста в определенных областях виджета `Text`.

11. Как осуществляется вставка виджетов в текстовое поле?

Вставка виджетов (например, кнопок или меток) в текстовое поле осуществляется с помощью метода `window_create()`. С помощью этого метода можно вставить объект `Tkinter` (например, виджет) в определенную позицию в тексте.

12. Для чего предназначены виджеты `Radiobutton` и `Checkbutton`?

`Radiobutton`: Предназначен для выбора одного из нескольких вариантов. Все радиокнопки в одной группе взаимодействуют друг с другом, и только одна из них может быть выбрана одновременно.

`Checkbutton`: Предназначен для выбора нескольких вариантов (можно поставить несколько флажков одновременно). Каждый флажок может быть как выбран, так и не выбран.

13. Что такое переменные `Tkinter` и для чего они нужны?

Переменные `Tkinter` (например, `StringVar`, `IntVar`, `BooleanVar`) – это особые объекты, которые используются для связи данных между виджетами и программой. Они позволяют отслеживать изменения в виджетах и обновлять их состояние.

14. Как осуществляется связь переменных `Tkinter` с виджетами `Radiobutton` и `Checkbutton`?

Для связи переменных с виджетами `Radiobutton` и `Checkbutton` используется параметр `variable`. Переменная хранит текущее состояние виджета (выбран/не выбран или значение радиокнопки).

Вывод: в ходе выполнения работы были приобретены навыки построения графического интерфейса пользователя GUI с помощью пакета Tkinter языка программирования Python версии 3.x.