

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 1**

Выполнил:  
Бабенко Артём Тимофеевич  
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:  
Ассистент департамента цифровых,  
робототехнических систем и  
электроники Богданов С.С

---

(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:

Задание 1. Решите задачу: 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".

Результат работы программы:



Рисунок 1 – Результат работы программы

Код программы:

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

from tkinter import *

def calculate(event):
    n1 = first_number.get()
    n2 = second_number.get()
    widget = event.widget
    if n1.isdigit() and n2.isdigit:
```

```

    if widget == summa:
        result = str(float(n1) + float(n2))
    elif widget == sub:
        result = str(float(n1) - float(n2))
    elif widget == multiply:
        result = str(float(n1) * float(n2))
    elif widget == divide:
        if n2 == 0:
            result = "Ошибка"
        else:
            result = str(float(n1) / float(n2))
    else:
        result = "Ошибка"
    answer.config(text=str(result))

if __name__ == '__main__':
    root = Tk()
    first_number = Entry(width=11)
    second_number = Entry(width=11)
    summa = Button(text="+", width=20)
    summa.bind("<Button-1>", calculate)
    sub = Button(text="-", width=20)
    sub.bind("<Button-1>", calculate)
    multiply = Button(text="*", width=20)
    multiply.bind("<Button-1>", calculate)
    divide = Button(text="/", width=20)
    divide.bind("<Button-1>", calculate)
    answer = Label(width=12,height=4, fg="black", bg="white", font="Arial 14")

    first_number.pack()
    second_number.pack()
    summa.pack()
    sub.pack()
    multiply.pack()
    divide.pack()
    answer.pack()

    root.mainloop()

```

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.

Результат работы программы:



Рисунок 2 – Результат работы программы

Код программы:

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

from tkinter import *

def color(color_name, color_code):
    answer.config(text=color_name)
    entry.delete(0, 'end')
    entry.insert(0, color_code)

if __name__ == '__main__':
    root = Tk()
    button_widths = 12
    entry = Entry(root, justify="center")
    answer = Label(root)
    red = Button(root, width=button_widths, bg='#ff0000', command=lambda: color("Красный",
"#ff0000"))
    orange = Button(root, width=button_widths, bg='#ff7d00', command=lambda:
color("Оранжевый", "#ff7d00"))
    yellow = Button(root, width=button_widths, bg='#ffff00', command=lambda:
color("Желтый", "#ffff00"))
    green = Button(root, width=button_widths, bg='#00ff00', command=lambda:
color("Зеленый", "#00ff00"))
    light_blue = Button(root, width=button_widths, bg='#007dff', command=lambda:
color("Голубой", "#007dff"))
    blue = Button(root, width=button_widths, bg='#0000ff', command=lambda: color("Синий",
"#0000ff"))
    purple = Button(root, width=button_widths, bg='#7d00ff', command=lambda:
color("Фиолетовый", "#7d00ff"))

    answer.pack()
    entry.pack()
    red.pack()
    orange.pack()
    yellow.pack()
```

```

green.pack()
light_blue.pack()
blue.pack()
purple.pack()

root.mainloop()

```

Задание 3. перепишите программу из пункта 2

Результат работы программы:

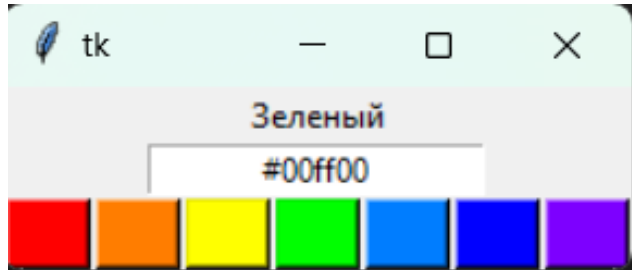


Рисунок 3 – Результат работы программы

Код программы:

```

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

from tkinter import *

def color(color_name, color_code):
    answer.config(text=color_name)
    entry.delete(0, 'end')
    entry.insert(0, color_code)

if __name__ == '__main__':
    root = Tk()
    button_widths = 3
    entry = Entry(root, justify="center")
    answer = Label(root)
    red = Button(root, width=button_widths, bg='#ff0000', command=lambda: color("Красный", "#ff0000"))
    orange = Button(root, width=button_widths, bg='#ff7d00', command=lambda: color("Оранжевый", "#ff7d00"))
    yellow = Button(root, width=button_widths, bg='#ffff00', command=lambda: color("Желтый", "#ffff00"))
    green = Button(root, width=button_widths, bg='#00ff00', command=lambda: color("Зеленый", "#00ff00"))
    light_blue = Button(root, width=button_widths, bg='#007dff', command=lambda: color("Голубой", "#007dff"))
    blue = Button(root, width=button_widths, bg='#0000ff', command=lambda: color("Синий", "#0000ff"))
    purple = Button(root, width=button_widths, bg='#7d00ff', command=lambda: color("Фиолетовый", "#7d00ff"))

```

```

answer.pack()
entry.pack()

button_frame = Frame(root)
button_frame.pack()
red.pack(side=LEFT, padx=1)
orange.pack(side=LEFT, padx=1)
yellow.pack(side=LEFT, padx=1)
green.pack(side=LEFT, padx=1)
light_blue.pack(side=LEFT, padx=1)
blue.pack(side=LEFT, padx=1)
purple.pack(side=LEFT, padx=1)

root.mainloop()

```

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, если указывать имена файлов без адреса.

Результат работы программы:



Рисунок 4 – Результат работы программы

Код программы:

```

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

from tkinter import *

def open_file():
    file_name = entry.get()
    if file_name:
        with open(file_name, 'r', encoding="utf-8") as file:
            content = file.read()

```

```

        text.delete(1.0, END)
        text.insert(1.0, content)

def save_file():
    file_name = entry.get()
    if file_name:
        with open(file_name, 'w', encoding="utf-8") as file:
            content = text.get(1.0, END).strip()
            file.write(content)

if __name__ == "__main__":
    root = Tk()
    frame1 = Frame(root)
    frame1.pack()
    entry = Entry(frame1, width=60)
    entry.pack(side=LEFT)
    open_button = Button(frame1, width=10,
                        text="Открыть", command=open_file)
    open_button.pack(side=LEFT)
    save_button = Button(frame1, width=10,
                        text="Сохранить", command=save_file)
    save_button.pack(side=LEFT)
    frame2 = Frame(root)
    frame2.pack(fill=BOTH, expand=True)

    x_scroll = Scrollbar(frame2, orient=HORIZONTAL)
    x_scroll.pack(side=BOTTOM, fill=X)

    y_scroll = Scrollbar(frame2, orient=VERTICAL)
    y_scroll.pack(side=RIGHT, fill=Y)
    text = Text(frame2, wrap=NONE, width=60, height=20,
                xscrollcommand=x_scroll.set, yscrollcommand=y_scroll.set)
    text.pack(side=LEFT, fill=BOTH, expand=True)
    x_scroll.config(command=text.xview)
    y_scroll.config(command=text.yview)
    root.mainloop()

```

Задание 5. Напишите программу, в которой имеется несколько объединенных в группу радиокнопок, индикатор которых выключен ( `indicatoron=0` ). Если какая-нибудь кнопка включается, то в метке должна отображаться соответствующая ей информация. Обычных кнопок в окне быть не должно.

Результат работы программы:

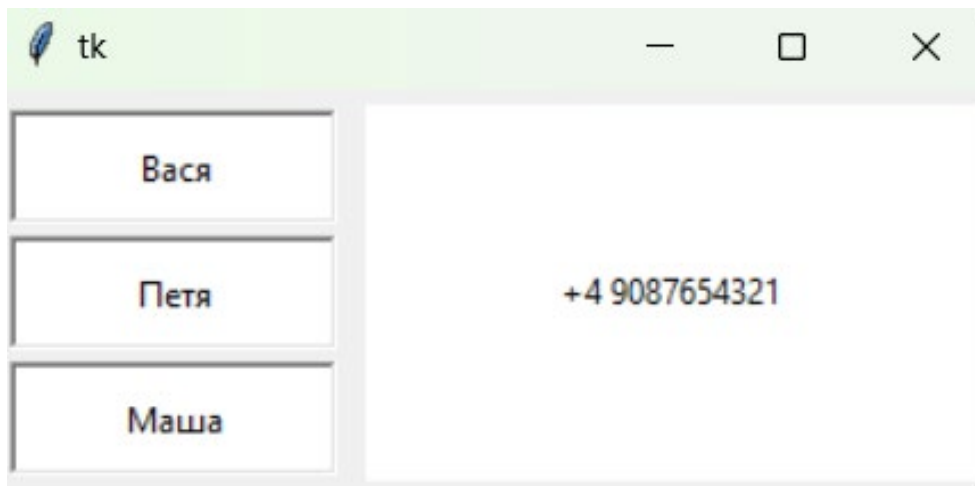


Рисунок 5 – Результат работы программы

Код программы:

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

from tkinter import *

def new_text(value):
    label.config(text=value)

if __name__ == "__main__":
    root = Tk()
    buttons = ["Вася", "Петя", "Маша"]
    numbers = ["888888888888", "+4 9087654321", "800000000000"]
    frame1 = Frame(root)
    frame1.pack(side=LEFT, padx=5, pady=5)
    for name, number in zip(buttons, numbers):
        Radiobutton(frame1, text=name, indicatoron=0, width=15, height=2,
                    command=lambda num=number: new_text(num)).pack(pady=2)
    frame2 = Frame(root, bg="white")
    frame2.pack(side=RIGHT, fill=BOTH, expand=True, padx=5, pady=5)
    label = Label(frame2, bg="white", width=30, anchor="center")
    label.pack(expand=True, fill=BOTH)
    root.mainloop()
```

Ответы на контрольные вопросы:

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

Tkinter, PyQt, wxPython, Kivy.

2. Что такое Tkinter?



Это стандартная библиотека Python для создания графических интерфейсов пользователя (GUI), основанная на Tcl/Tk.

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

- 1) Создать главное окно.
- 2) Создать виджеты и выполнить конфигурацию их свойств (опций).
- 3) Определить события, то есть то, на что будет реагировать программа.
- 4) Описать обработчики событий, то есть то, как будет реагировать программа.
- 5) Расположить виджеты в главном окне.
- 6) Запустить цикл обработки событий.
4. Что такое цикл обработки событий?

Это механизм, который ждет и обрабатывает события (например, нажатия кнопок, ввод текста) в приложении.

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

Он представляет главное окно приложения и управляет его поведением, а также взаимодействием с системой.

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

Button: Кнопка для выполнения действий.

Label: Текстовая метка для отображения информации.

Entry: Однострочное текстовое поле для ввода данных.

Text: Многострочное текстовое поле для ввода и отображения текста.

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

Он используется для автоматического размещения виджетов в окне, упрощая процесс их компоновки.

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

Можно использовать параметры `side`, `fill`, `expand` для управления расположением и размерами виджетов.

9. Как осуществляется управление полосами прокрутки в виджете `Text`?

Полосы прокрутки создаются с помощью виджета `Scrollbar`, который связывается с виджетом `Text` через методы `config()` и `set()`.

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

Они позволяют выделять части текста (например, изменять цвет или шрифт) и управлять стилем отображения.

11. Как осуществляется вставка виджетов в текстовое поле?

В `Tkinter` нельзя вставлять виджеты напрямую в `Text`, но можно использовать фреймы или другие контейнеры для организации интерфейса.

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

`Radiobutton`: Позволяет выбрать один вариант из нескольких.

`Checkbutton`: Позволяет выбрать несколько вариантов (включать/выключать).

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

Это специальные переменные (`StringVar`, `IntVar` и т.д.), которые связывают данные с виджетами и автоматически обновляют их при изменении значений.

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

Эти виджеты используют переменные `Tkinter` для хранения состояния (выбран/не выбран) и обновляют их при взаимодействии пользователя

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