

МИНИСТЕРСТВО НАУКИ И ВЫСШЕГО ОБРАЗОВАНИЯ  
РОССИЙСКОЙ ФЕДЕРАЦИИ  
Федеральное государственное автономное  
образовательное учреждение высшего образования  
«СЕВЕРО-КАВКАЗСКИЙ ФЕДЕРАЛЬНЫЙ УНИВЕРСИТЕТ»

Кафедра инфокоммуникаций

Отчет по лабораторной работе №1

Основы работы с Tkinter

По дисциплине «Объектно-ориентированное программирование»

Выполнил студент группы ИВТ-б-о-20-1

Новикова В.И. « » \_\_\_\_\_ 20\_\_ г.

Подпись студента \_\_\_\_\_

Работа защищена « » \_\_\_\_\_ 20\_\_ г.

Проверил Воронкин Р. А. \_\_\_\_\_

(подпись)

Ставрополь 2022

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

**Ход работы:**

1. Создала общедоступный репозиторий и клонировала на его на локальный сервер.

[https://github.com/Valentina1502/LAB\\_4\\_7.git](https://github.com/Valentina1502/LAB_4_7.git)

2. После изучения теоретического материала и методических рекомендаций приступила к выполнению заданий.

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"

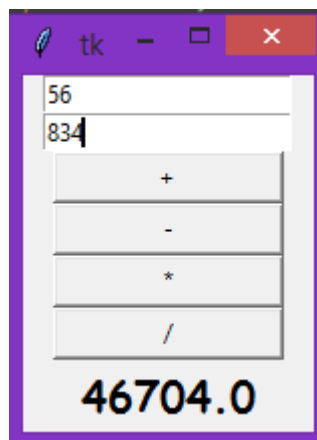


Рисунок 1 – Задание 1

Код:

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

from tkinter import *

def addition(event):
    try:
        s1 = float(e1.get())
        s2 = float(e2.get())
        l1['text'] = s1+s2
```

```

except ValueError:
    l1['text'] = 'Ошибка'

def subtraction(event):
    try:
        s1 = float(e1.get())
        s2 = float(e2.get())
        l1['text'] = s1-s2
    except ValueError:
        l1['text'] = 'Ошибка'

def multiply(event):
    try:
        s1 = float(e1.get())
        s2 = float(e2.get())
        l1['text'] = s1*s2
    except ValueError:
        l1['text'] = 'Ошибка'

def devide(event):
    try:
        s1 = float(e1.get())
        s2 = float(e2.get())
        l1['text'] = s1/s2
    except ValueError:
        l1['text'] = 'Ошибка'
    except ZeroDivisionError:
        l1['text'] = 'Деление на 0!'

if __name__ == '__main__':
    root = Tk()
    e1 = Entry(width=20)
    e2 = Entry(width=20)
    b1 = Button(text='+', width=15)
    b2 = Button(text='-', width=15)
    b3 = Button(text='*', width=15)
    b4 = Button(text='/', width=15)
    l1 = Label(font=(
        "Comic Sans MS",
        16,
        "bold"
    ))
    b1.bind('<Button-1>', addition)
    b2.bind('<Button-1>', subtraction)
    b3.bind('<Button-1>', multiply)
    b4.bind('<Button-1>', devide)
    l1.config(bd=10)
    e1.pack()
    e2.pack()
    b1.pack()
    b2.pack()
    b3.pack()
    b4.pack()
    l1.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.

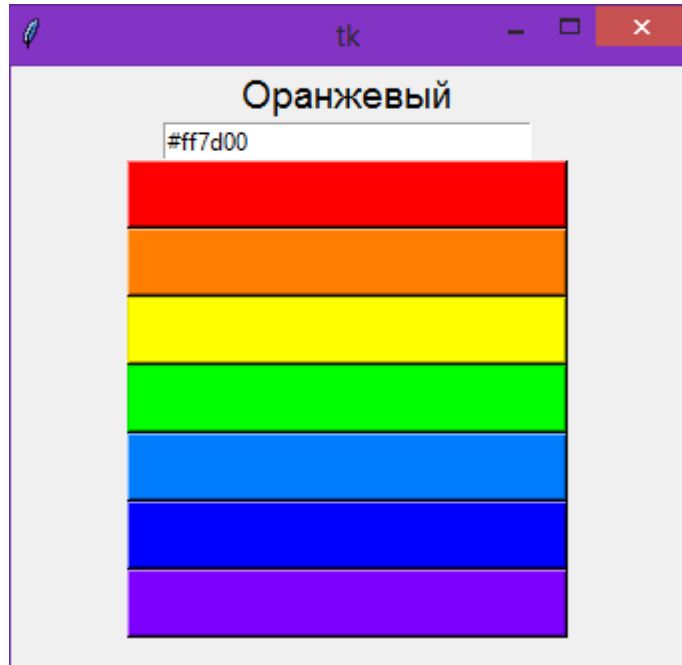


Рисунок 2 – Задание 2

Код:

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

from tkinter import *

def red(event):
    ent1.delete(0, END)
    l1['text'] = "Красный"
    ent1.insert(0, "#ff0000")

def orange(event):
    ent1.delete(0, END)
    l1['text'] = 'Оранжевый'
    ent1.insert(0, "#ff7d00")

def yellow(event):
    ent1.delete(0, END)
    l1['text'] = 'Жёлтый'
    ent1.insert(0, "#ffff00")

def green(event):
    ent1.delete(0, END)
    l1['text'] = 'Зелёный'
    ent1.insert(0, "#00ff00")

def gol(event):
```

```

ent1.delete(0, END)
l1['text'] = 'Голубой'
ent1.insert(0, "#007dff")

def blue(event):
    ent1.delete(0, END)
    l1['text'] = 'Синий'
    ent1.insert(0, "#0000ff")

def fiol(event):
    ent1.delete(0, END)
    l1['text'] = 'Фиолетовый'
    ent1.insert(0, "#7d00ff")

if __name__ == '__main__':
    root = Tk()
    l1 = Label(font="Arial 14", width=30)
    ent1 = Entry(width=30)
    but1 = Button(bg='#ff0000', width=30, pady=5)
    but2 = Button(bg='#ff7d00', width=30, pady=5)
    but3 = Button(bg='#ffff00', width=30, pady=5)
    but4 = Button(bg='#00ff00', width=30, pady=5)
    but5 = Button(bg='#007dff', width=30, pady=5)
    but6 = Button(bg='#0000ff', width=30, pady=5)
    but7 = Button(bg='#7d00ff', width=30, pady=5)
    but1.bind('<Button-1>', red)
    but2.bind('<Button-1>', orange)
    but3.bind('<Button-1>', yellow)
    but4.bind('<Button-1>', green)
    but5.bind('<Button-1>', gol)
    but6.bind('<Button-1>', blue)
    but7.bind('<Button-1>', fiol)
    l1.pack()
    ent1.pack()
    but1.pack()
    but2.pack()
    but3.pack()
    but4.pack()
    but5.pack()
    but6.pack()
    but7.pack()
    root.mainloop()

```

Задание 3. 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

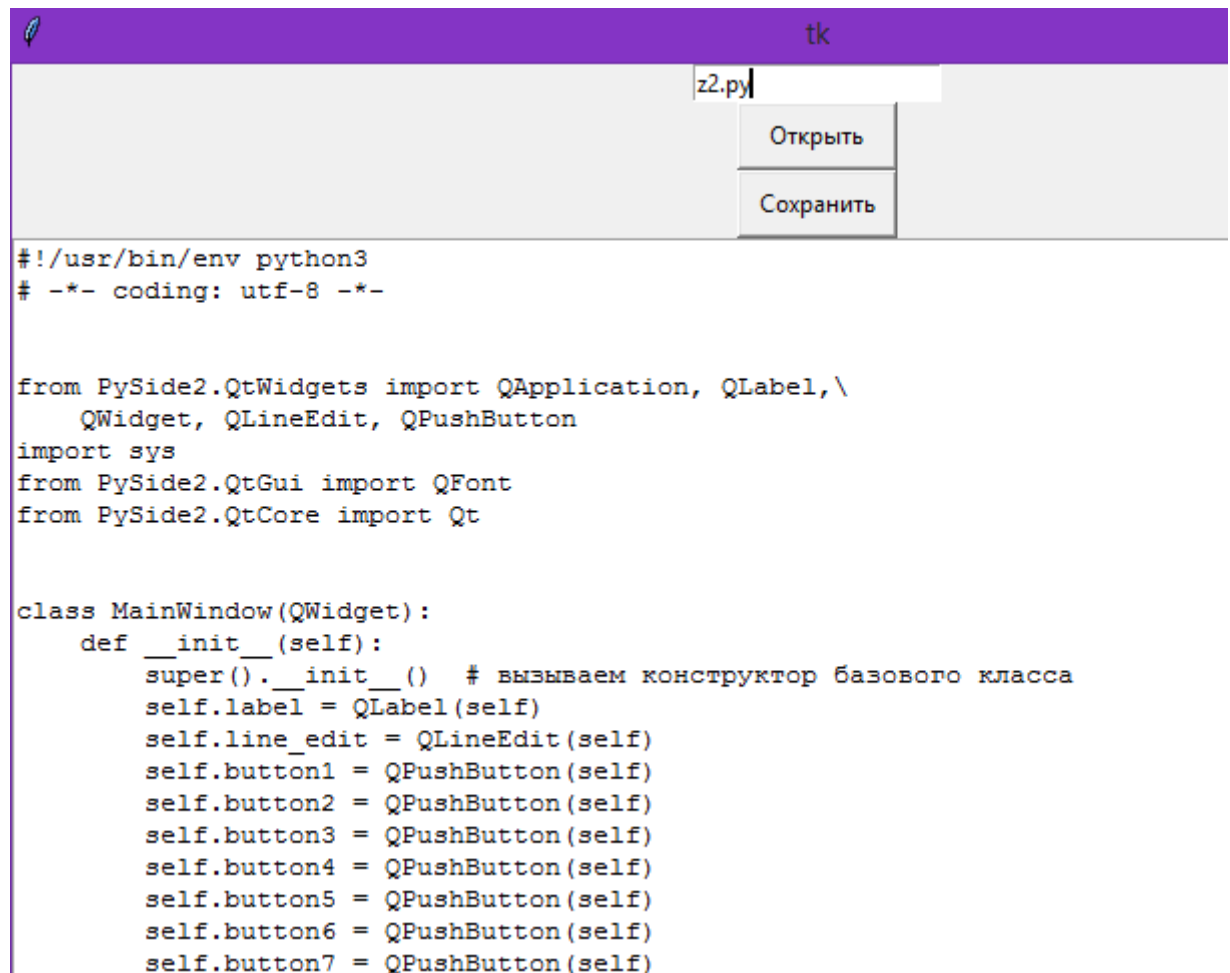


Рисунок 3 – Задание 3

Код:

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

from tkinter import Tk, Text, Button, Entry, END, filedialog

def save():
    name = filedialog.asksaveasfilename(defaultextension=".txt",
    filetypes=(("Текстовый файл", "*.txt"),))
    data = text.get(1.0, END)
    with open(name, 'w', encoding="utf-8") as f:
        f.write(data)

def opening():
    text.delete(1.0, END)
    name = filedialog.askopenfilename()
    with open(name, 'r', encoding="utf-8") as f:
        data = f.read()
    text.insert(1.0, data)

if __name__ == '__main__':
    root = Tk()
    text = Text(width=100, height=80,)
    ent = Entry(width=20)
    but1 = Button(text='Открыть', width=10, pady=5, command=opening)
```

```

but2 = Button(text='Сохранить', width=10, pady=5, command=save)
ent.pack()
but1.pack()
but2.pack()
text.pack()
root.mainloop()

```

Задание 4. 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

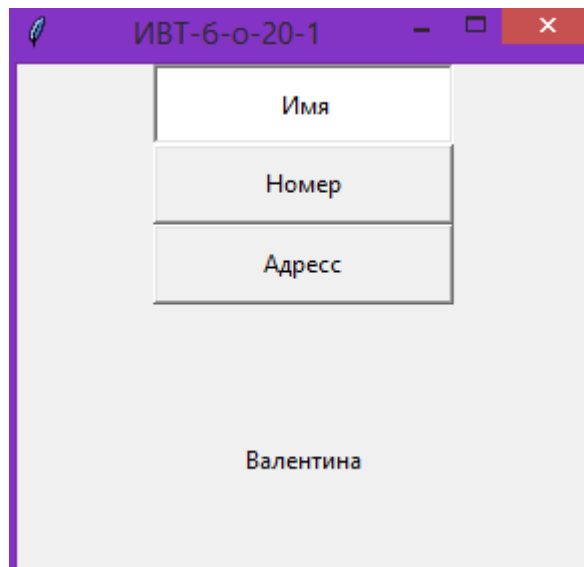


Рисунок 4 – Задание 4

Код:

```

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

from tkinter import Tk, IntVar, Radiobutton, Label

def name():
    label['text'] = 'Валентина'

def number():
    label['text'] = '8-999-743-23-25'

def adress():
    label['text'] = 'Ставрополь'

if __name__ == '__main__':
    root = Tk()
    root.title('ИВТ-6-о-20-1')
    var = IntVar()
    var.set(-1)
    Radiobutton(indicatoron=0, text="Имя", command=name, variable=var,

```

```
value=0, width=20, height=2).pack()  
Radiobutton(indicatoron=0, text="Номер", command=number, variable=var,  
value=1, width=20, height=2).pack()  
Radiobutton(indicatoron=0, text="Адресс", command=address, variable=var,  
value=2, width=20, height=2).pack()  
label = Label(width=40, height=25)  
label.pack()  
root.mainloop()
```

### Контрольные вопросы:

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

Существует множество библиотек GUI, среди которых Tk не самый популярный инструмент, хотя с его помощью написано немало проектов. Он был выбран для Python по-умолчанию. Установочный файл интерпретатора Питона обычно уже включает пакет tkinter в составе стандартной библиотеки.

2. Что такое Tkinter?

Tkinter – это пакет для Python, предназначенный для работы с библиотекой Tk. Библиотека Tk содержит компоненты графического интерфейса пользователя (graphical user interface – GUI), написанные на языке программирования Tcl.

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

– Создать главное окно. – Создать виджеты и выполнить конфигурацию их свойств (опций). – Определить события, то есть то, на что будет реагировать программа. – Описать обработчики событий, то есть то, как будет реагировать программа. – Расположить виджеты в главном окне. – Запустить цикл обработки событий.

4. Что такое цикл обработки событий?

Tkinter является событийно-ориентированной библиотекой. В приложениях такого типа имеется главный цикл обработки событий. В Tkinter такой цикл запускается методом `mainloop`. Для явного выхода из интерпретатора и завершения цикла обработки событий используется метод `quit`.



5. Каково назначение экземпляра класса Tk при построении графического интерфейса с помощью Tkinter? Метод `mainloop` экземпляра Tk запускает главный цикл обработки событий, что в том числе приводит к отображению главного окна со всеми "упакованными" на нем виджетами

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

Button – кнопка. Самыми важными свойствами виджета класса Button являются `text`, с помощью которого устанавливается надпись на кнопке, и `command` для установки действия, то есть того, что будет происходить при нажатии на кнопку. Label – метка Виджет Label просто отображает текст в окне и служит в основном для информационных целей (вывод сообщений, подпись других элементов интерфейса). Entry – однострочное текстовое поле. Текстовые поля предназначены для ввода информации пользователем.

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

Метод «`Pack()`» - упаковщик. Ранее мы его уже использовали для отображения наших виджетов в окне программы, но использовали без дополнительных параметров. И по умолчанию наши виджеты располагались друг под другом, в зависимости в какой последовательности был применен метод «`pack()`» к объектам.

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

Если в упаковщики не передавать аргументы, то виджеты будут располагаться вертикально, друг над другом. Тот объект, который первым вызовет `pack`, будет сверху. Который вторым – под первым, и так далее. У метода `pack` есть параметр `side` (сторона), который принимает одно из четырех значений-констант `tkinter` – TOP, BOTTOM, LEFT, RIGHT (верх, низ, лево, право). По умолчанию, когда в `pack` не указывается `side`, его значение равняется TOP. Из-за этого виджеты располагаются вертикально.

9. Как осуществляется управление полосами прокрутки в виджете Text?

Если в текстовое поле вводится больше линий текста, чем его высота, то оно само будет прокручиваться вниз. При просмотре прокручивать вверх-вниз можно с помощью колеса мыши и стрелками на клавиатуре. Однако бывает удобнее пользоваться скроллером – полосой прокрутки. В tkinter скроллеры производятся от класса `Scrollbar`. Объект-скроллер связывают с виджетом, которому он требуется. Это не обязательно многострочное текстовое поле.

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

Особенностью текстового поля библиотеки Tk является возможность форматировать текст в нем, то есть придавать его разным частям разное оформление. Делается это с помощью методов `tag_add` и `tag_config`. Первый добавляет тег, при этом надо указать его произвольное имя и отрезок текста, к которому он будет применяться. Метод `tag_config` настраивает тегу стили оформления.

11. Как осуществляется вставка виджетов в текстовое поле?

В `Text` можно вставлять другие виджеты помощью метода `window_create`. Потребность в этом не велика, однако может быть интересна с объектами типа `Canvas`.

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

`Checkbutton` – это виджет, который позволяет отметить „галочкой“ определенный пункт в окне. При использовании нескольких пунктов нужно каждому присвоить свою переменную; `Radiobutton` выполняет функцию, схожую с функцией виджета `Checkbutton`. Разница в том, что в виджете `Radiobutton` пользователь может выбрать лишь один из пунктов.

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

В Tkinter нельзя использовать любую переменную для хранения состояний виджетов. Для этих целей предусмотрены специальные классы-переменные пакета tkinter – `BooleanVar`, `IntVar`, `DoubleVar`, `StringVar`. Первый класс позволяет принимать своим экземплярам только булевы значения (0 или 1 и `True` или `False`), второй – целые, третий – дробные, четвертый – строковые.

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

При запуске программы включенной окажется первая радиокнопка, так как значение ее опции value совпадает с текущим значением переменной `r_var`. Если кликнуть по второй радиокнопке, то она включится, а первая выключится. При этом значение `r_var` станет равным 1. В функции `change` в зависимости от считанного значения переменной `var` ход выполнения программы идет по одной из трех веток.

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