ФГБОУ ВО Чувашский государственный

университет имени И.Н. Ульянова

**Лабораторная работа № 7**

**"GUI, классы, модуль Tkinter"**Вариант № 11

Выполнил студент ЭЭ-21-21

Михайлов Николай Алексеевич

Чебоксары 2022

**Цели работы:**

Дать студентам практический навык в написании программ, использующих графический интерфейс пользователя (GUI). Ознакомить с такими понятиями, как класс, виджеты, массивы записей, а так же с атрибутами класса и методами передачи данных между классами.

**Краткие теоретические сведения:**

Модуль Tkinter:

*https://docs.python.org/3/library/tkinter.html*

**Рабочее задание:**

1.1 Описать запись с именем Route, содержащую следующие поля:

* название начального пункта маршрута;
* название конечного пункта маршрута;
* номер маршрута.

1.2 Написать программу, выполняющую следующие действия:

ввод данных с клавиатуры в массив, состоящий из восьми элементов типа Route; записи

должны быть упорядочены по номерам маршрутов;

* вывод на экран информации о маршрутах, которые начинаются или кончаются в пункте, название которого введено с клавиатуры; если таких маршрутов нет, вывести на экран соответствующее сообщение;
* запись массива в файл под заданным с клавиатуры именем.

**import** tkinter **as** tk

**from** tkinter **import** filedialog **as** fd

**from** tkinter **import** messagebox **as** mbox

**class** App():

**def** **\_\_init\_\_**(self, window):

**self.**window = window

        # *Создание главного меню*

        mainmenu = tk**.Menu**(**self.**window)

**self.**window**.config**(menu=mainmenu)

        # *Создание каскадного меню "Файл"*

        filemenu = tk**.Menu**(mainmenu, tearoff=0)

        filemenu**.add\_command**(label = "Создать", command=**self.**new\_file)

        filemenu**.add\_command**(label = "Открыть..", command=**self.**open\_file)

        # *Инициализация каскадного меню*

        mainmenu**.add\_cascade**(label="Файл", menu=filemenu)

        # *Навзвание открытого файла*

**self.**this\_file = tk**.Label**(window, text=" ", font='Arial 10')

**self.**this\_file**.grid**(row=0, column=0, columnspan=2, sticky='W', padx=10)

        # *пункт и поиск*

**self.**point = tk**.Entry**(**self.**window, width= 20)

**self.**point**.grid**(row=1, column=0, padx=10)

**self.**find = tk**.Button**(**self.**window, text='Найти', command=**self.**find\_point,)

**self.**find**.grid**(row=1, column=1, padx=70, ipadx=10)

        # *Запуск приложения*

**self.**window**.mainloop**()

**def** **open\_file**(self):

**self.**filename = fd**.askopenfilename**(

            title="Открыть файл",

            initialdir="/",

            filetypes=(

                ("Текстовый файл", "\*.txt"),

                )

            )

        # *Вывод названия открытого файла*

**self.**this\_file['text'] = **self.**filename**.split**('/')[-1]

**def** **find\_point**(self):

        # *Обработка исключения*

**if** **self.**this\_file['text'] **==** ' ':

            mbox**.showerror**("Ошибка", "Файл не выбран")

**return**

        # *Обзор файла*

**self.**points = []

**self.**rout\_names = []

**with** **open**(f'{**self.**filename}', encoding='utf-8') **as** w\_file:

**for** row **in** w\_file:

                data = row**.split**(';')

**self.**points**.append**(data[:2])

**self.**rout\_names**.append**(data[2][:-1])

        required\_point = **self.**point**.get**()

**if** **not** required\_point**.isalpha**():

            mbox**.showerror**("Ошибка", "Укажите название пункта")

**elif** required\_point **not** **in** **sum**(**self.**points, []):

            mbox**.showinfo**("Нет маршрута", "Через данный пункт не проходит ни один маршрут")

**else**:

            tk**.Label**(window, text='Проходят маршруты:')**.grid**(row=2, column=0, columnspan=2, sticky='W', padx=10)

            row\_index = 3

            i = 0

**for** pair\_points, route\_name **in** **zip**(**self.**points, **self.**rout\_names):

**if** required\_point **in** pair\_points:

                    this\_rout = tk**.Label**(window, text=' ,'[i%2] + route\_name)

                    this\_rout**.grid**(row=row\_index, column=i%2, padx=10, sticky='W')

**if** i%2 **==** 1:

                        row\_index += 1

                    i += 1

**def** **new\_file**(self):

        # *Создание дочернего окна*

**self.**new\_file = tk**.Toplevel**(master=window)

**self.**new\_file**.resizable**(width=**False**, height=**False**)

**self.**new\_file**.title**('Создать файл')

**self.**new\_file**.grab\_set**()

        # *создание рамки, где будут распологать поля ввода*

        frame = tk**.Frame**(**self.**new\_file)

        tk**.Label**(frame, text='начало')**.grid**(row=0, column=1, padx=3)

        tk**.Label**(frame, text='конец')**.grid**(row=0, column=2, padx=3)

        tk**.Label**(frame, text='номер')**.grid**(row=0, column=3, padx=3)

        # *табуляция всех строк*

        tabs = [tk**.Label**(frame, text=f'{i}') **for** i **in** **range**(1, 9)]

        # *создание всех полей ввода начального пункта*

**self.**start\_wid = [tk**.Entry**(frame) **for** \_ **in** **range**(1, 9)]

        # *создание всех полей ввода конечного пункта*

**self.**finish\_wid = [tk**.Entry**(frame) **for** \_ **in** **range**(1, 9)]

        # *создание всех полей ввода номера маршрута*

**self.**number\_wid = [tk**.Entry**(frame, width=10) **for** \_ **in** **range**(1, 9)]

        # *размещение Entry и их табуляции*

**for** i **in** **range**(1, 9):

            tabs[i-1]**.grid**(row=i, column=0, pady=5, padx=3)

**self.**start\_wid[i-1]**.grid**(row=i, column=1, pady=5, padx=3)

**self.**finish\_wid[i-1]**.grid**(row=i, column=2, pady=5, padx=3)

**self.**number\_wid[i-1]**.grid**(row=i, column=3, pady=5, padx=3)

        # *упаковка рамки*

        frame**.pack**(padx=5, pady=2)

        # *сохдание меню сохранения*

        save = tk**.Menu**(**self.**new\_file)

**self.**new\_file**.config**(menu=save)

        # *кнопка сохранения*

        save**.add\_command**(label='Сохранить', command=**self.**save\_file)

**def** **save\_file**(self):

**self.**start\_data = [data**.get**() **for** data **in** **self.**start\_wid]

**self.**finish\_data = [data**.get**() **for** data **in** **self.**finish\_wid]

**self.**number\_data = [data**.get**() **for** data **in** **self.**number\_wid]

        new\_file = fd**.asksaveasfile**(

            title="Сохранить файл",

            defaultextension=".txt",filetypes=(("Текстовый файл", "\*.txt"),))

**with** new\_file:

**for** i **in** **range**(8):

                new\_file**.write**(f'{**self.**start\_data[i]};{**self.**finish\_data[i]};{**self.**number\_data[i]}\n')

**self.**new\_file**.destroy**()

**self.**filename =  new\_file**.**name

**self.**this\_file['text'] = **self.**filename**.split**('/')[-1]

**if** \_\_name\_\_ **==** "\_\_main\_\_":

    window = tk**.Tk**()

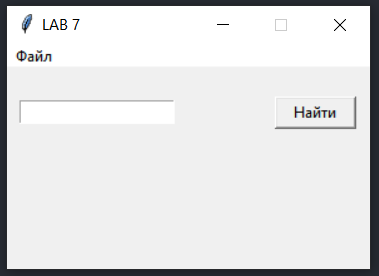
    window**.title**('LAB 7')

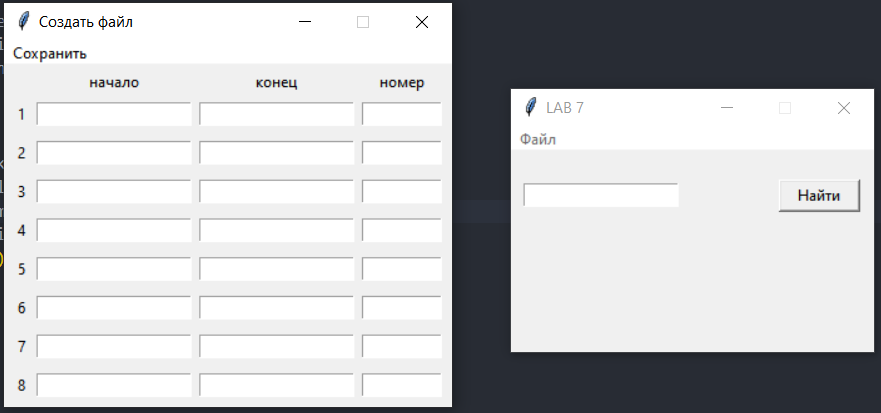
    window**.geometry**('290x160+500+200')

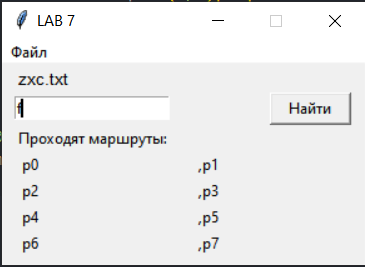
    window**.resizable**(width=**False**, height=**False**)

**App**(window)

**Результат:**







**Выводы:**

При выполнении работы был затронут модуль Tkinter, элементы ООП. Реализован графический интерфейс для программы