

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 №8
дисциплины «Объектно-ориентированное программирование»

Выполнил:
Кожуховский Виктор Андреевич
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 Роман Александрович

(подпись)

Отчет защищен с оценкой _____ Дата защиты _____

Ставрополь, 2024 г.

Тема: Обработка событий и рисование в Tkinter

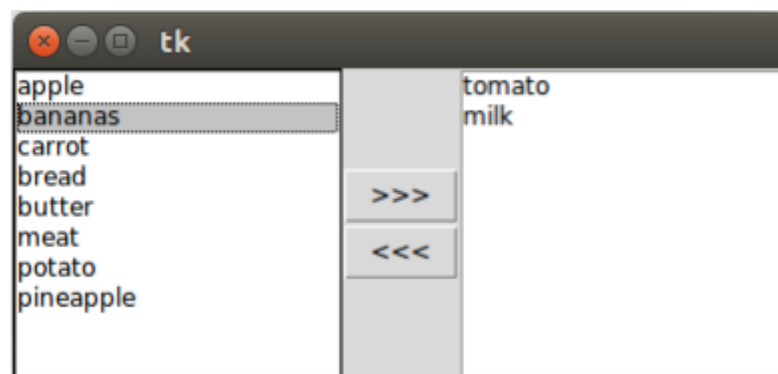
Цель: приобретение навыков улучшения графического интерфейса пользователя GUI с помощью обработки событий и рисования, реализованных в пакете Tkinter языка программирования Python версии 3.x.

Порядок выполнения работы:

1. Изучил теоретический материал работы.
2. Создал общедоступный репозиторий на GitHub, в котором использована лицензия MIT и язык программирования Python.
3. Выполнил клонирование созданного репозитория.
4. Дополнил файл .gitignore необходимыми правилами для работы с IDE.
5. Организовал свой репозиторий в соответствии с моделью ветвления git-flow.
6. Создал проект в папке репозитория.
7. Проработал примеры лабораторной работы.
8. Выполнил индивидуальное задание для варианта 14.

Задание 1

Напишите программу, состоящую из двух списков Listbox . В первом будет, например, перечень товаров, заданный программно. Второй изначально пуст, пусть это будет перечень покупок. При клике на одну кнопку товар должен переходить из одного списка в другой. При клике на вторую кнопку – возвращаться (человек передумал покупать). Предусмотрите возможность множественного выбора элементов списка и их перемещения.



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

from tkinter import Tk, Frame, Listbox, Button, END, EXTENDED

def move_to_cart() -> None:
    selected_indices = goods_list.curselection()

    for index in reversed(selected_indices):
        item = goods_list.get(index)
        cart_list.insert(END, item)
        goods_list.delete(index)

def remove_from_cart() -> None:
    selected_indices = cart_list.curselection()

    for index in reversed(selected_indices):
        item = cart_list.get(index)
        goods_list.insert(END, item)
        cart_list.delete(index)

root = Tk()

frame = Frame(root)
frame.pack()

goods_list = Listbox(frame, width=30, height=10, selectmode=EXTENDED)
goods_list.grid(row=1, column=0)

goods = [
    "apple",
    "bananas",
    "carrot",
    "bread",
    "butter",
    "meat",
    "potato",
    "pineapple",
    "milk",
    "tomato",
]

for item in goods:
    goods_list.insert(END, item)

button_frame = Frame(frame)
button_frame.grid(row=1, column=1)

to_cart_btn = Button(button_frame, text=">>>", command=move_to_cart)
to_cart_btn.pack()

from_cart_btn = Button(button_frame, text="<<<", command=remove_from_cart)
from_cart_btn.pack()

cart_list = Listbox(frame, width=30, height=10, selectmode=EXTENDED)
cart_list.grid(row=1, column=2)

root.mainloop()
```

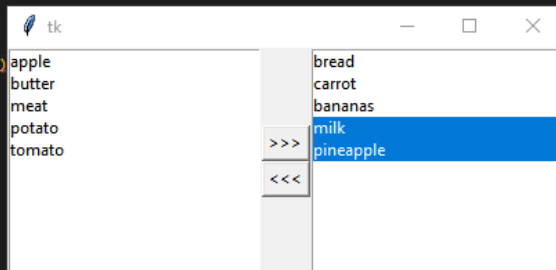


Рисунок 1. Код решения индивидуального задания 1 и его выполнение

Задание 2

Напишите программу по следующему описанию. Нажатие Enter в однострочном текстовом поле приводит к перемещению текста из него в список (экземпляр Listbox). При двойном клике (<Double-Button-1>) по элементу-строке списка, она должна копироваться в текстовое поле.

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

from tkinter import Tk, Frame, Listbox, END, EXTENDED, W, X, BOTH, Entry, Label

def add_to_list(event: None = None) -> None:
    text = entry.get().strip()

    if text:
        list_box.insert(END, text)
        entry.delete(0, END)

def copy_to_entry(event: None) -> None:
    try:
        selected_index = list_box.curselection()[0]
        selected_text = list_box.get(selected_index)

        entry.delete(0, END)
        entry.insert(0, selected_text)
    except IndexError:
        pass

root = Tk()

frame = Frame(root)
frame.pack(pady=10, padx=10, fill=BOTH, expand=True)

label_entry = Label(frame, text="Введите текст:")
label_entry.pack(anchor=W)

entry = Entry(frame, width=50)
entry.pack(fill=X, pady=5)
entry.bind("<Return>", add_to_list)

label_list = Label(frame, text="Список:")
label_list.pack(anchor=W)

list_box = Listbox(frame, width=50, height=10, selectmode=EXTENDED)
list_box.pack(fill=BOTH, expand=True)

list_box.bind("<Double-Button-1>", copy_to_entry)

root.mainloop()
```

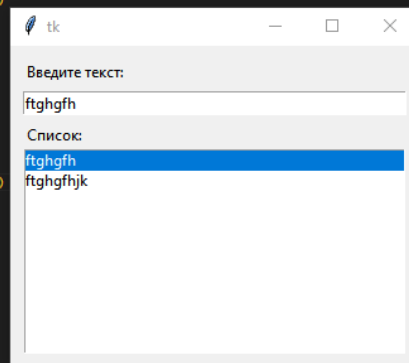
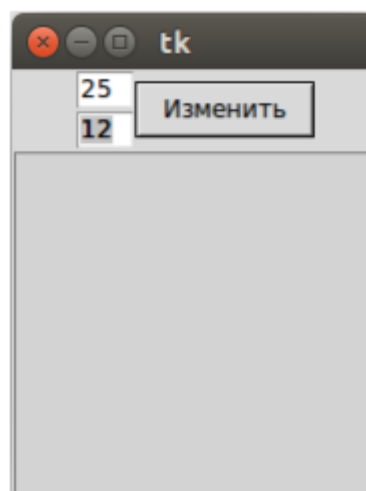


Рисунок 2. Код решения индивидуального задания 2 и его выполнение

Задание 3

напишите программу по описанию. Размеры многострочного текстового поля определяются значениями, введенными в однострочные текстовые поля. Изменение размера происходит при нажатии мышью на кнопку, а также при нажатии клавиши Enter. Цвет фона экземпляра Text светлосерый (lightgrey), когда поле не в фокусе, и белый, когда имеет фокус. Событие получения фокуса обозначается как <FocusIn>, потери – как <FocusOut>. Для справки: фокус перемещается по виджетам при нажатии Tab, Ctrl+Tab, Shift+Tab, а также при клике по ним мышью (к кнопкам последнее не относится).



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

from tkinter import Tk, Entry, Button, Text
from typing import Optional, Any

def change_text_size(event: Optional[Any] = None) -> None:
    try:
        width = int(width_entry.get())
        height = int(height_entry.get())
        text_widget.config(width=width, height=height)
    except ValueError:
        pass

def focus_in(event: Any) -> None:
    event.widget.config(bg="white")

def focus_out(event: Any) -> None:
    event.widget.config(bg="lightgrey")

root: Tk = Tk()
root.title("Изменение размера текстового поля")

width_entry: Entry = Entry(root)
width_entry.pack()
width_entry.bind("<Return>", change_text_size)

height_entry: Entry = Entry(root)
height_entry.pack()
height_entry.bind("<Return>", change_text_size)

change_button: Button = Button(
    root,
    text="Изменить размер",
    command=change_text_size,
)
change_button.pack()

text_widget: Text = Text(root, width=30, height=10, bg="lightgrey")
text_widget.pack()
text_widget.bind("<FocusIn>", focus_in)
text_widget.bind("<FocusOut>", focus_out)

root.mainloop()
```

Рисунок 3. Код решения индивидуального задания 1 и его выполнение

Задание 4

Создайте на холсте подобное изображение:

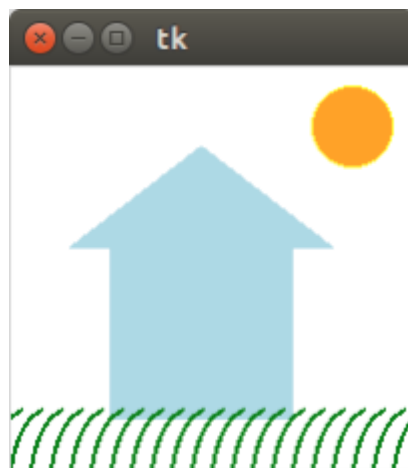




Рисунок 4. Код решения индивидуального задания 2 и его выполнение
Задание 5

В данной программе создается анимация круга, который движется от левой границы холста до правой:

```
from tkinter import *
def motion():
    c.move(ball, 1, 0)
    if c.coords(ball)[2] < 300:
        root.after(10, motion)
root = Tk()
c = Canvas(root, width=300, height=200,
            bg="white")
c.pack()
ball = c.create_oval(0, 100, 40, 140,
                    fill='green')
motion()
root.mainloop()
```

Выражение `c.coords(ball)` возвращает список текущих координат объекта (в данном случае это `ball`). Третий элемент списка соответствует его второй координате `y`. Метод `after` вызывает функцию, переданную вторым аргументом, через количество миллисекунд, указанных первым аргументом.

Изучите приведенную программу и самостоятельно запрограммируйте постепенное движение фигуры в ту точку холста, где пользователь кликает левой кнопкой мыши. Координаты события хранятся в его атрибутах `x` и `y` (`event.x`, `event.y`).

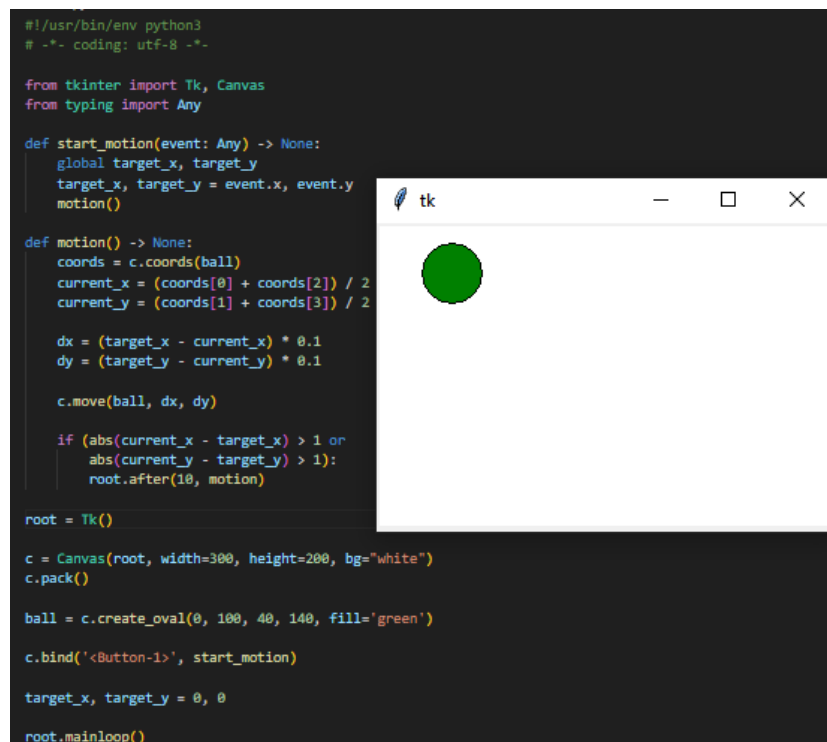


Рисунок 5. Код решения индивидуального задания 1 и его выполнение

9. Зафиксировал сделанные изменения в репозитории.
10. Выполнил слияние ветки для разработки с веткой `master/main`.
11. Отправил сделанные изменения на сервер GitHub.

Ссылка: <https://github.com/Viktorkozh/OOP-8>

Контрольные вопросы:

1. Каково назначение виджета `ListBox`?

Назначение виджета `ListBox`: "От класса `ListBox` создаются списки – виджеты, внутри которых в столбик перечисляются элементы. При этом можно выбирать один или множество элементов списка."

2. Каким образом осуществляется связывание событие или действие с виджетом Tkinter?

Связывание события или действия с виджетом Tkinter осуществляется с помощью метода "bind": "В tkinter с помощью метода bind между собой связываются виджет, событие и действие."

3. Какие существуют типы событий в Tkinter? Приведите примеры.

Существуют три основных типа событий в Tkinter: "производимые мышью, нажатиями клавиш на клавиатуре, а также события, возникающие в результате изменения виджетов." Примеры событий: "<Button-1>" – клик левой кнопкой мыши, "<Return>" – нажатие клавиши Enter.

4. Как обрабатываются события в Tkinter?

Обработка событий в Tkinter осуществляется через привязку функции-обработчика к событию с помощью метода "bind": "При вызове метода bind событие передается в качестве первого аргумента."

5. Как обрабатываются события мыши в Tkinter?

События мыши в Tkinter обрабатываются с помощью привязки к ним функций-обработчиков: "Часто используемые события, производимые мышью: <Button-1> – клик левой кнопкой мыши, <Button-2> – клик средней кнопкой мыши, <Button-3> – клик правой кнопкой мыши."

6. Каким образом можно отображать графические примитивы в Tkinter?

Графические примитивы в Tkinter отображаются на холсте с помощью методов, создающих объекты на экземпляре Canvas: "При создании экземпляра Canvas необходимо указать его ширину и высоту."

7. Перечислите основные методы для отображения графических примитивов в Tkinter.

Основные методы для отображения графических примитивов в Tkinter: "create_line", "create_rectangle", "create_polygon", "create_oval", "create_arc", "create_text".

8. Каким образом можно обратиться к ранее созданным фигурам на холсте?

К ранее созданным фигурам на холсте можно обратиться с помощью идентификаторов и тегов: "Методы, создающие фигуры на холсте, возвращают численные идентификаторы этих объектов, которые можно присвоить переменным."

9. Каково назначение тэгов в Tkinter?

Назначение тэгов в Tkinter: "Один и тот же тег может присваиваться разным объектам. Дальнейшее обращение к такому тегу позволит изменить все объекты, в которых он был указан."

Вывод: приобрел навыки улучшения графического интерфейса пользователя GUI с помощью обработки событий и рисования, реализованных в пакете Tkinter языка программирования Python версии 3.x.