

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

**Кафедра инфокоммуникаций
«Обработка событий и рисование в Tkinter»**

**Отчет по лабораторной работе № 4.8
по дисциплине «Объектно-ориентированное программирование»**

Выполнил студент группы ИВТ-б-о-21-1

Богадунов Василий Игоревич.

« » _____ 20__г.

Подпись студента _____

Работа защищена « » _____ 20__г.

Проверил Воронкин Р.А. _____
(подпись)

Цель работы: 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. Создал общедоступный репозиторий на GitHub, в котором использована лицензия MIT и язык программирования Python.

Create a new repository

A repository contains all project files, including the revision history. Already have a project repository elsewhere? [Import a repository.](#)

Required fields are marked with an asterisk (*).

Owner * Repository name *

 ItsMyLife1337 /

Great repository names are short and memorable. Need inspiration? How about [friendly-system](#) ?

Description (optional)



- ☒  **Public**
Anyone on the internet can see this repository. You choose who can commit.
- ☐  **Private**
You choose who can see and commit to this repository.

Рисунок 1 – Создание репозитория

2. Выполнение клонирования созданного репозитория.

```
c:\Users\Admin\Desktop\git>git clone https://github.com/ItsMyLife1337/OOP_4.8.git
Cloning into 'OOP_4.8'...
remote: Enumerating objects: 5, done.
remote: Counting objects: 100% (5/5), done.
remote: Compressing objects: 100% (5/5), done.
remote: Total 5 (delta 0), reused 0 (delta 0), pack-reused 0
Receiving objects: 100% (5/5), done.
```

Рисунок 2 – Клонирование репозитория

3. Организация репозитория в соответствии с моделью ветвления git-flow.

```

c:\Users\Admin\Desktop\git\OOP_4.8>git flow init

Which branch should be used for bringing forth production releases?
- main
Branch name for production releases: [main]
Branch name for "next release" development: [develop]

How to name your supporting branch prefixes?
Feature branches? [feature/]
Bugfix branches? [bugfix/]
Release branches? [release/]
Hotfix branches? [hotfix/]
Support branches? [support/]
Version tag prefix? []
Hooks and filters directory? [C:/Users/Admin/Desktop/git/OOP_4.8/.git/hooks]

```

Рисунок 3 – Ветвление по модели git-flow

Задание №1. Решите задачу: 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.



Рисунок 4 – Пример и результат выполнения задания

Задание 2. Решите задачу: 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.

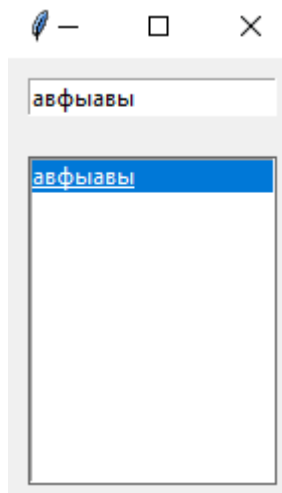


Рисунок 5 – Результат выполнения задания

Задание 3. Решите задачу: 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).



Рисунок 6 – Результат выполнения

Задание 4. Решите задачу: Создайте на холсте подобное изображение (для создания травы используется цикл):

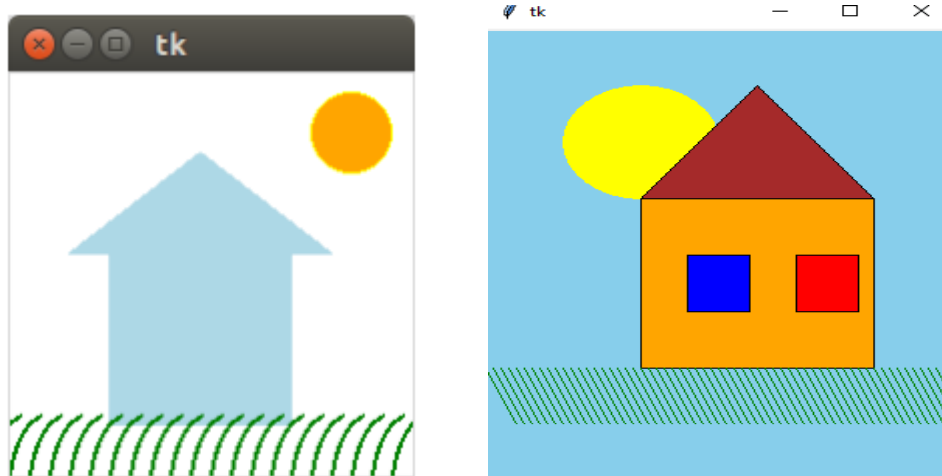


Рисунок 7 – Слева пример, справа результат выполнения задания

Задание №5. Решите задачу: 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 `x`. 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).

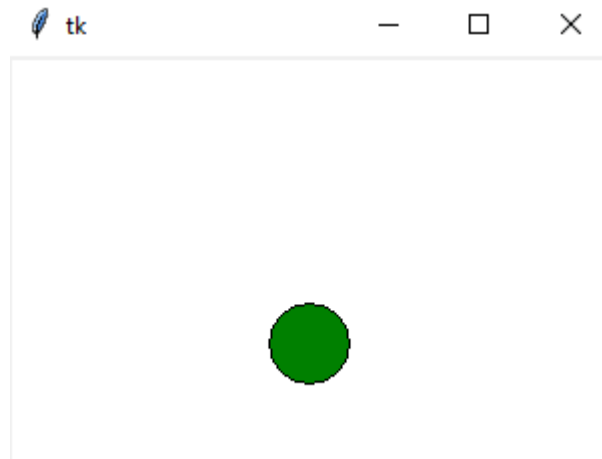


Рисунок 9 – Результат выполнения

Вывод: в результате выполнения лабораторной работы были приобретены навыки улучшения графического интерфейса пользователя GUI с помощью обработки событий и рисования, реализованных в пакете Tkinter языка программирования Python версии 3.x.

Ответы на контрольные вопросы:

1. Каково назначение виджета ListBox?

Виджет ListBox в Tkinter предназначен для отображения списка элементов, из которых пользователь может выбирать один или несколько. Этот виджет предоставляет прокручиваемый список элементов.

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

Связывание событий в Tkinter осуществляется с использованием метода bind(). Этот метод позволяет привязывать функции-обработчики к определенным событиям, таким как нажатие кнопки, перемещение мыши и другие.

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

Существует множество типов событий в Tkinter. Некоторые из них включают:

Button-1: Левая кнопка мыши.

Button-2: Средняя кнопка мыши (если она есть).

Button-3: Правая кнопка мыши.

Motion: Движение мыши.

KeyPress: Нажатие клавиши на клавиатуре.

KeyRelease: Отпускание клавиши на клавиатуре.

4. Как обрабатываются события в Tkinter?

События обрабатываются путем привязки функций-обработчиков к определенным событиям с использованием метода `bind()` или через метод `after()` для периодического выполнения действий.

5. Как обрабатываются события мыши в Tkinter?

События мыши, такие как нажатие кнопок или движение мыши, обрабатываются путем привязки функций-обработчиков к соответствующим событиям с использованием метода `bind()`.

6. Каким образом можно отображать графические примитивы в Tkinter?

Для отображения графических примитивов в Tkinter используется виджет `Canvas`. Этот виджет позволяет рисовать линии, прямоугольники, окружности и другие графические элементы.

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

Некоторые основные методы для работы с графическими примитивами на холсте (`Canvas`):

`create_line()`: Создает линию.

`create_rectangle()`: Создает прямоугольник.

`create_oval()`: Создает овал.

`create_text()`: Создает текст.

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

Каждая фигура на холсте имеет уникальный идентификатор, который возвращается методами создания фигур. Идентификаторы могут использоваться для обращения к ранее созданным фигурам.

9. Каково назначение тэгов в Tkinter?

Тэги в Tkinter используются для группировки и идентификации набора объектов на холсте. Они позволяют применять действия к определенным группам объектов. Тэги могут быть присвоены при создании объекта на холсте или позднее с использованием метода `addtag_withtag()`.