

Лекция 7

Создание приложений с графическим интерфейсом пользователя

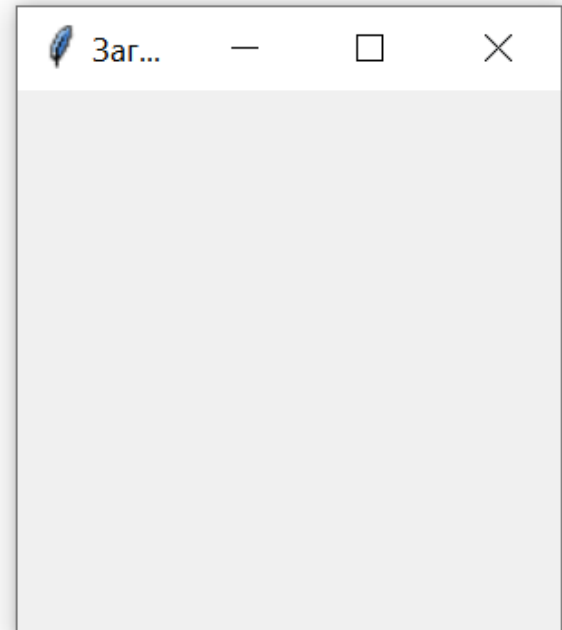
Библиотека tkinter

```
from tkinter import *
```

Виджеты – это базовые элементы для создания графического интерфейса программы.

Создание окна:

```
from tkinter import *  
  
Window = Tk()  
Window.title("Заголовок окна")  
  
Window.mainloop()
```



| Метод | Описание | Пример |
|---|--|--|
| <code>title("Заголовок")</code> | заголовок окна | <code>Window.title("Заголовок окна")</code> |
| <code>geometry('ШxB+x+y')</code>) | размер и положение окна | <code>Window.geometry('400x300+200+100')</code> <code>Window.geometry('400x300')</code> <code>Window.geometry('+300+300')</code> <code>Window.geometry('-0-0')</code> |
| <code>update_idletasks()</code> | перезагрузка данных об окне | <code>Window.update_idletasks()</code> <code>s = Window.geometry()</code> |
| <code>winfo_screenwidth()</code> | ширина экрана (не окна) | <code>Window = Tk()</code> # ширина и высота экрана <code>w = Window.winfo_screenwidth()</code> <code>h = Window.winfo_screenheight()</code> <code>w = w//2</code> # середина экрана <code>h = h//2</code> <code>w = w - 200</code> # смещение от середины <code>h = h - 200</code> # расположение формы по центру <code>Window.geometry('400x400+{ }+{ }'.format(w, h))</code> |
| <code>winfo_screenheight()</code> | высота экрана (не окна) | |
| <code>resizable(По_горизон тали, По_вертикали)</code> | запрет изменения размеров главного окна по горизонтали и/или вертикали | <code>Window.resizable(False, False)</code> <code>Window.resizable(False, True)</code> <code>Window.resizable(1, 0)</code> |
| <code>mainloop()</code> | отображение окна и запуск цикла обработки событий окна для взаимодействия с пользователем | <code>Window.mainloop()</code> |

Создание виджетов:

**объект = конструктор (родительский_виджет =
главное_окно_приложения, font , bg , fg , text , width , height ,
command , ...)**

font – шрифт

bg – цвет фона

fg – цвет надписи

width – ширина

height - высота

text - надпись

command – команда, выполняющаяся при активации виджета

Пример:

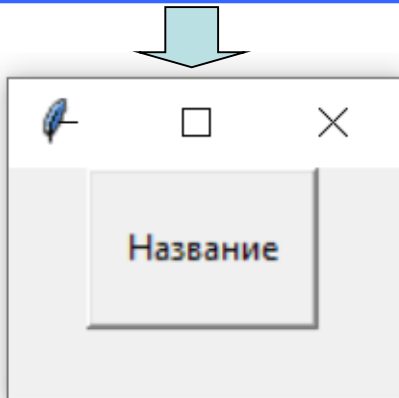
```
l1 = Label (bg='green', text="3")
```

Изменение атрибутов виджетов:

```
from tkinter import *

Window = Tk()
Window.geometry('130x80')
b1 = Button(text="Название", \
            width=10, height=3)

b1.pack()
Window.mainloop()
```



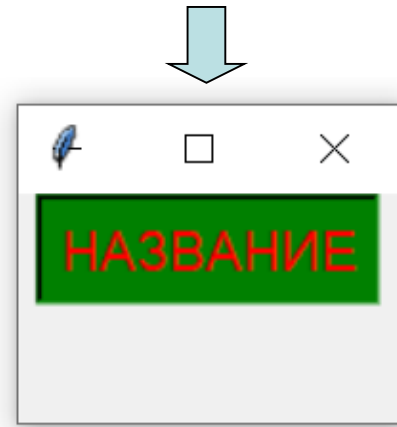
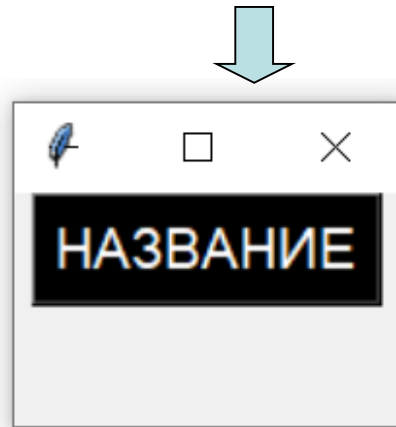
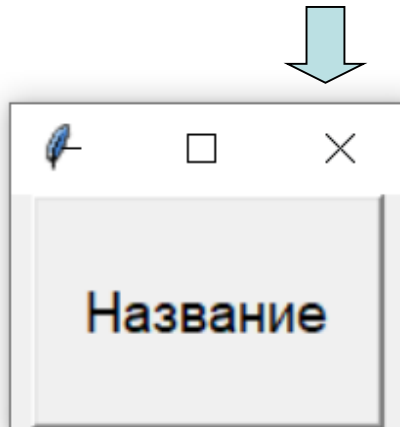
```
from tkinter import *

Window = Tk()
Window.geometry('130x80')
b1 = Button(text="Название", \
            width=10, height=3)

def click():
    b1['text'] = "НАЗВАНИЕ"
    b1['width'] = 10
    b1['height'] = 1
    b1['bg'] = '#000000'
    b1['activebackground'] = 'green'
    b1['fg'] = '#ffffff'
    b1['activeforeground'] = 'red'

b1.config (font = ("bold",14), command=click)

b1.pack()
Window.mainloop()
```



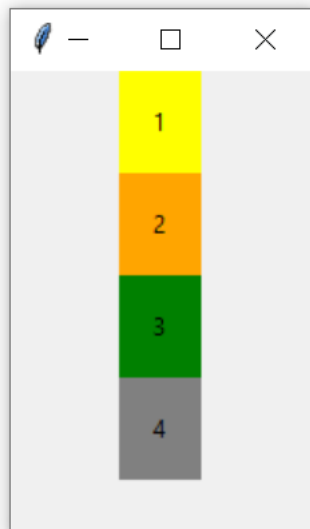
Отображение и расположение виджетов:

1) **pack (side, fill, expand, anchor)** – автоматическое размещение виджетов в родительском окне (**side** – сторона, **fill** – заполнение, **expand** – расширение, **anchor** - якорь)

side = TOP, BOTTOM, LEFT, RIGHT (верх, низ, лево, право); по умолчанию TOP

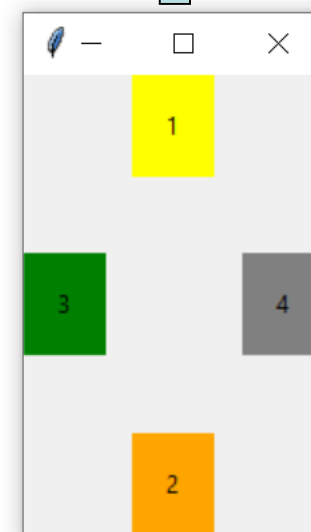
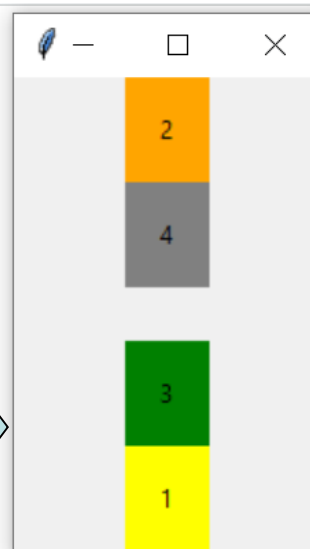
```
from tkinter import *  
  
Window = Tk()  
Window.geometry('150x230')  
  
l1 = Label(width=5, height=3, bg='yellow', text="1")  
l2 = Label(width=5, height=3, bg='orange', text="2")  
l3 = Label(width=5, height=3, bg='green', text="3")  
l4 = Label(width=5, height=3, bg='grey', text="4")
```

```
l1.pack(side = TOP)  
l2.pack(side = BOTTOM)  
l3.pack(side = LEFT)  
l4.pack(side = RIGHT)
```



```
l1.pack()  
l2.pack()  
l3.pack()  
l4.pack()
```

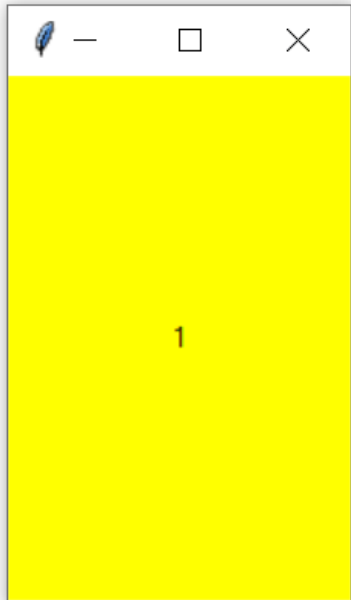
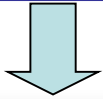
```
l1.pack(side = BOTTOM)  
l2.pack()  
l3.pack(side = BOTTOM)  
l4.pack()
```



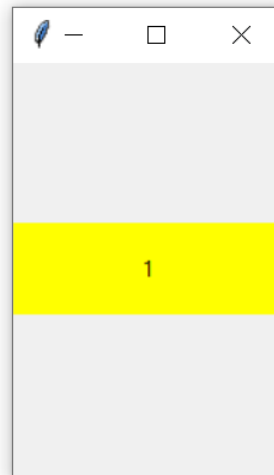
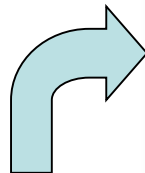
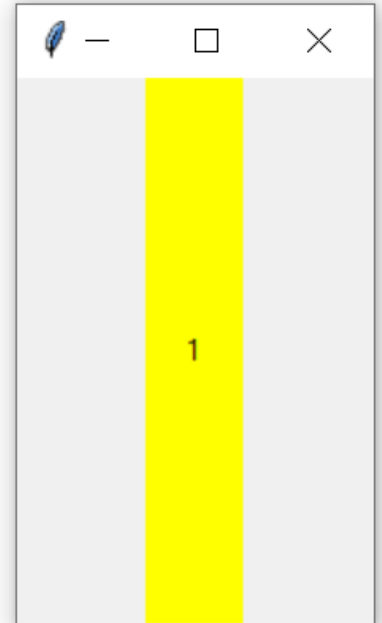
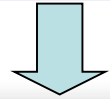
fill = NONE, BOTH, X, Y (заполнение отключено, заполнить все доступное пространство, заполнение по оси X, заполнение по оси Y);
по умолчанию NONE

expand = 0, 1 (0 – отключено; 1 – виджет всегда в середине);
по умолчанию 0

```
l1 = Label(width=5, height=3, \
            bg='yellow', text="1")
l1.pack(fill=BOTH, expand=1)
```



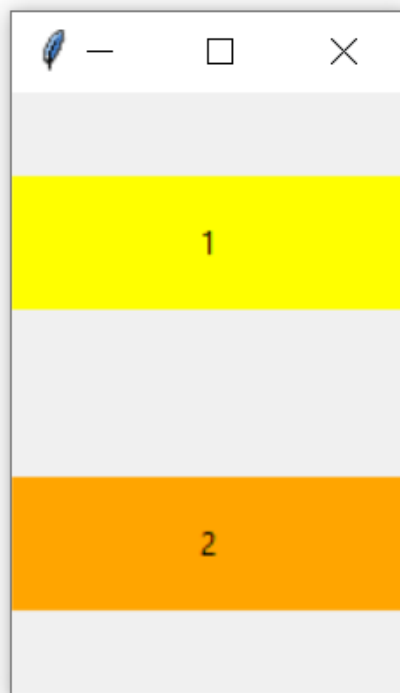
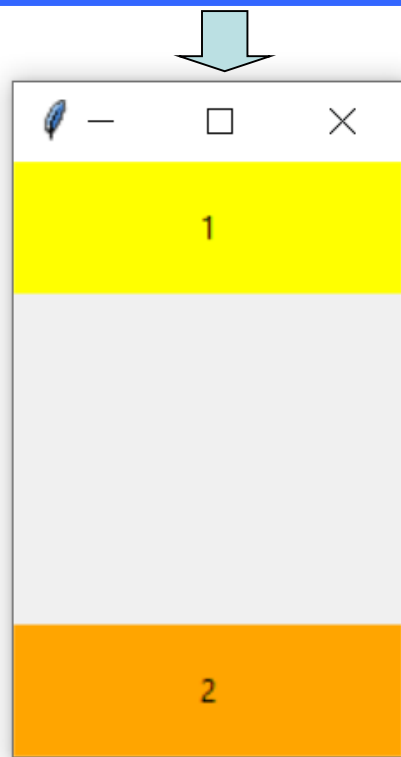
```
l1 = Label(width=5, height=3, \
            bg='yellow', text="1")
l1.pack(fill=Y, expand=1)
```



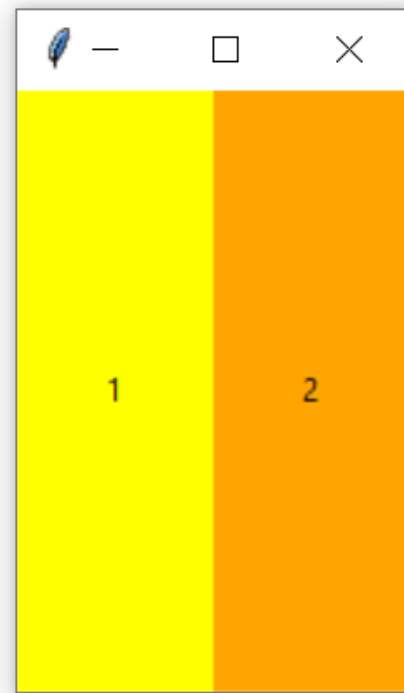
```
l1 = Label(width=5, height=3, \
            bg='yellow', text="1")
l1.pack(fill=X, expand=1)
```

```
l1 = Label(width=5, height=3, bg='yellow', text="1")  
l2 = Label(width=5, height=3, bg='orange', text="2")
```

```
l1.pack(side = TOP, fill = X)  
l2.pack(side = BOTTOM, fill = X)
```



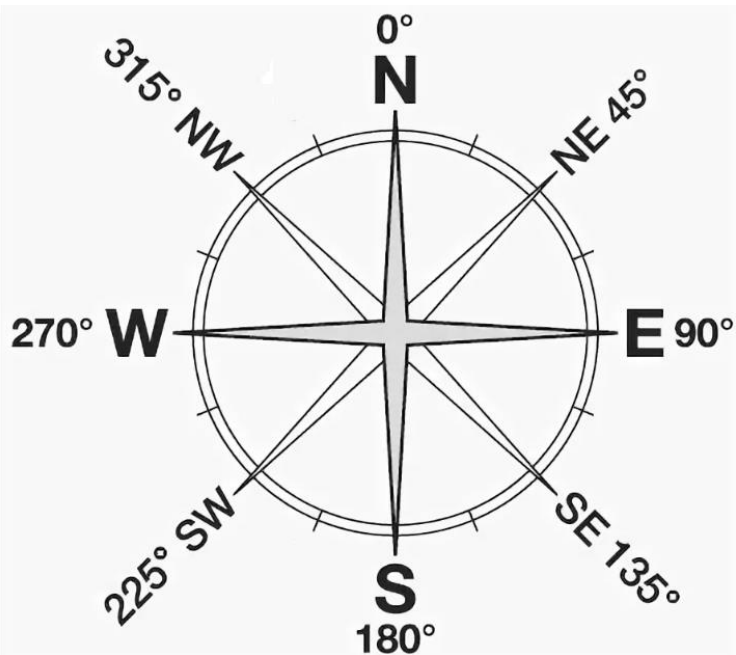
```
l1.pack(side = TOP, fill = X, expand = 1)  
l2.pack(side = BOTTOM, fill = X, expand = 1)
```



```
l1.pack(side = LEFT, fill = BOTH, expand = 1)  
l2.pack(side = LEFT, fill = BOTH, expand = 1)
```

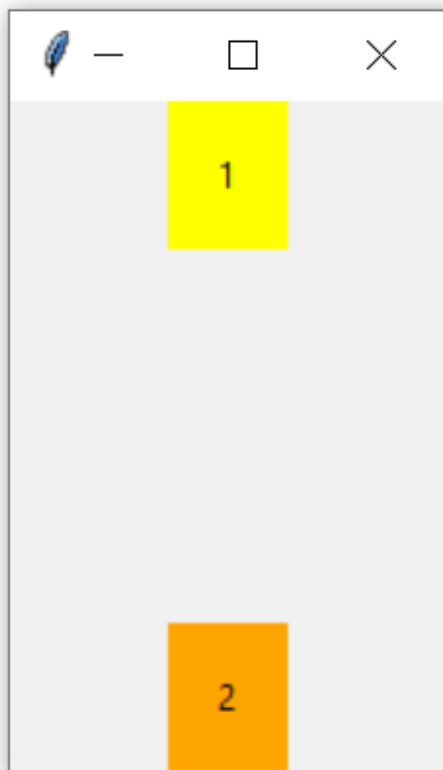

anchor = N (north – север), S (south – юг), W (west – запад), E (east – восток) и комбинации

```
l1 = Label(width=5, height=3, bg='yellow', text="1")  
l2 = Label(width=5, height=3, bg='orange', text="2")
```



```
l1.pack(side = TOP, anchor = NE)  
l2.pack(side = BOTTOM, anchor = SW)
```

```
l1.pack(side = TOP)  
l2.pack(side = BOTTOM)
```



2) **grid (*row, column, rowspan, colspan*)** – размещает виджетов на сетке (***row*** – строка в сетке, ***column*** – столбец в сетке, ***rowspan*** – сколько строк занимает виджет, ***colspan*** – сколько столбцов занимает виджет)

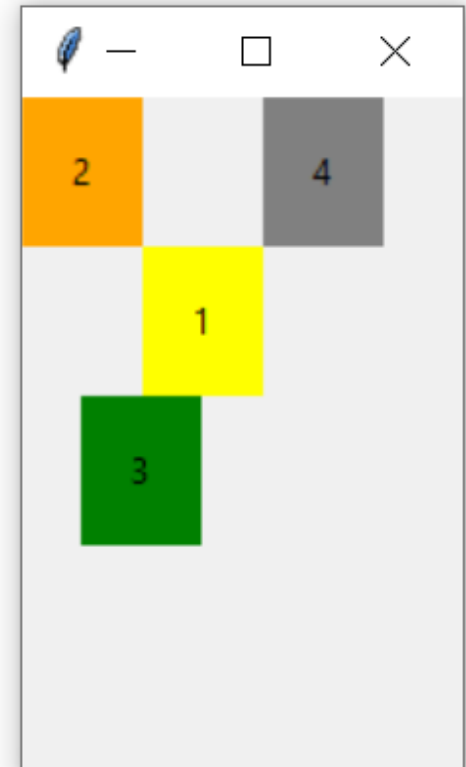
```
from tkinter import *

Window = Tk()
Window.geometry('150x230')

l1 = Label(width=5, height=3, bg='yellow', text="1")
l2 = Label(width=5, height=3, bg='orange', text="2")
l3 = Label(width=5, height=3, bg='green', text="3")
l4 = Label(width=5, height=3, bg='grey', text="4")

l1.grid(row = 1, column = 1)
l2.grid(row = 0, column = 0)
l3.grid(row = 2, column = 0, colspan = 2)
l4.grid(row = 0, column = 2)

Window.mainloop()
```



3) place (*x, y, width, height*) – размещение виджетов в указанных координатах с указанными размерами.

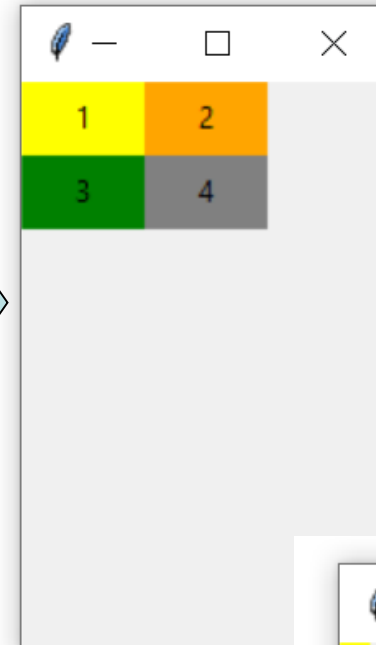
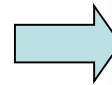
```
from tkinter import *

Window = Tk()
Window.geometry('150x230')

l1 = Label(bg='yellow', text="1")
l2 = Label(bg='orange', text="2")
l3 = Label(bg='green', text="3")
l4 = Label(bg='grey', text="4")

l1.place(x = 0, y = 0, width = 50, height = 30)
l2.place(x = 50, y = 0, width = 50, height = 30)
l3.place(x = 0, y = 30, width = 50, height = 30)
l4.place(x = 50, y = 30, width = 50, height = 30)

Window.mainloop()
```

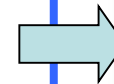


```
from tkinter import *

Window = Tk()
Window.geometry('150x230')

l1 = Label(bg='yellow', text="1").place(x = 0, y = 0)
l2 = Label(bg='orange', text="2").place(x = 0, y = 50)
l3 = Label(bg='green', text="3").place(x = 40, y = 0, width = 100)
l4 = Label(bg='grey', text="4").place(x = 20, y = 0, height = 100)

Window.mainloop()
```



Создание надписи (Label):

```
from tkinter import *

window = Tk()
window.title("GUI")
window.geometry('350x150')

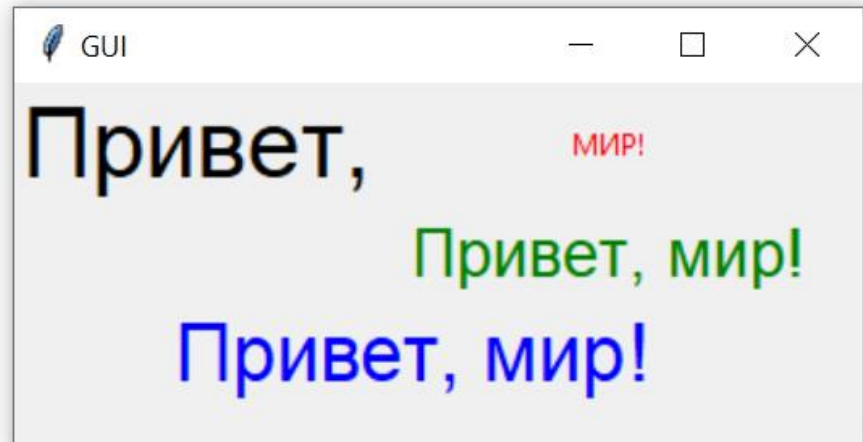
L1 = Label(window, text="Привет, ", font=("Arial Bold", 30))
L1.grid(column=0, row=0)

L2 = Label(window, text="МИР!", fg = 'red')
L2.grid(column=1, row=0)

L3 = Label(window, text="Привет, мир!", font=("Times New Roman Bold", 20), fg = 'green')
L3.grid(column=1, row=1)

L4 = Label(window, text="Привет, мир!", font=("Times New Roman Bold", 25), fg = 'blue')
L4.grid(column=0, row=2, columnspan = 3)

window.mainloop()
```



Создание текстового поля (Entry):

объект = Entry (родительский_виджет = главное_окно_приложения, bg, bd, cursor, fg, justify, relief, selectbackground, selectforeground, show, state, textvariable, width)

bg – фоновый цвет

bd – толщина границы

cursor – курсор указателя мыши при наведении на текстовое поле

fg – цвет текста

font – шрифт текста

justify – устанавливает выравнивание текста (LEFT, CENTER, RIGHT)

relief – определяет тип границы, по умолчанию значение FLAT

selectbackground – фоновый цвет выделенного куска текста

selectforeground – цвет выделенного текста

show – задает маску для вводимых символов

state – состояние элемента (NORMAL, DISABLED)

textvariable – устанавливает привязку к элементу StringVar

width – ширина элемента

```
from tkinter import *

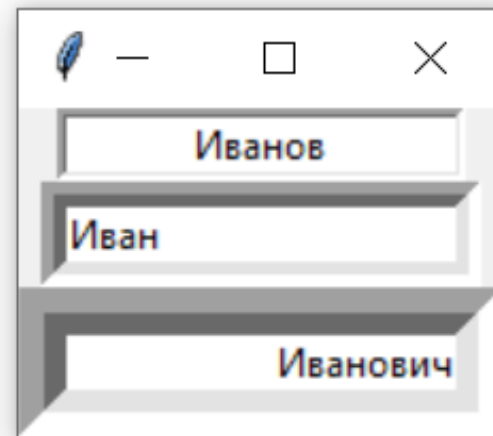
def click():
    lb3.configure(text = ed1.get())

window = Tk()
ed1 = Entry(window, width=20, bd = 3, justify = CENTER)
ed1.grid(column=0, row=0)

ed2 = Entry(window, width=20, bd = 8, justify = LEFT)
ed2.grid(column=0, row=1)

ed3 = Entry(window, width=20, bd = 15, justify = RIGHT)
ed3.grid(column=0, row=2)

ed2.focus()
window.mainloop()
```



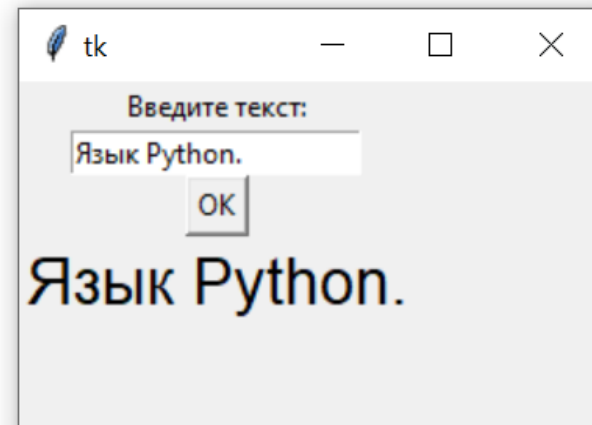
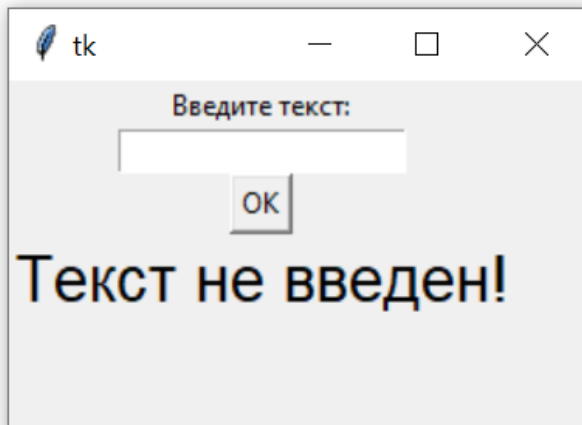
Создание кнопки (Button):

```
from tkinter import *

def click():
    lb3.configure(text = ed1.get())

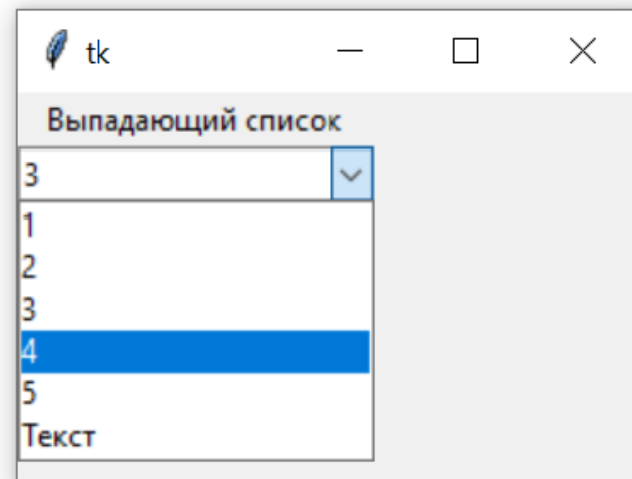
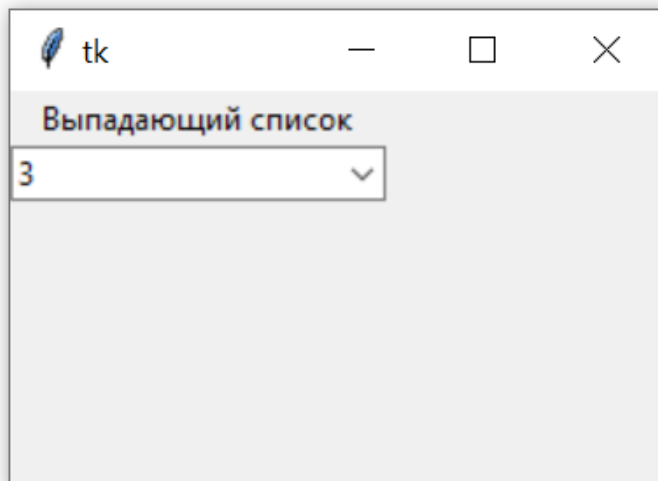
window = Tk()
window.geometry('250x150')
lbl = Label(window, text="Введите текст:")
lbl.grid(column=0, row=0)
ed1 = Entry(window, width=20)
ed1.grid(column=0, row=1)
btn1 = Button(window, text="OK", command = click)
btn1.grid(column=0, row=2)
lb3 = Label(window, text="Текст не введен!", font=("Arial Bold", 20))
lb3.grid(column=0, row=3)

window.mainloop()
```



Создание выпадающего списка (Combobox):

```
from tkinter import *  
from tkinter.ttk import Combobox  
  
window = Tk()  
window.geometry('250x150')  
  
lbl = Label(window, text="Выпадающий список")  
lbl.grid(column=0, row=0)  
  
combo = Combobox(window)  
combo['values'] = (1, 2, 3, 4, 5, "Текст")  
  
# установка варианта по умолчанию  
combo.current(2)  
  
combo.grid(column=0, row=1)  
window.mainloop()
```



Создание флажков (Checkbutton):

```
from tkinter import *
from tkinter.ttk import Checkbutton

window = Tk()
window.geometry('250x150')

chk1 = Checkbutton(window, text='Флажок №1')
chk1.grid(column=0, row=0)

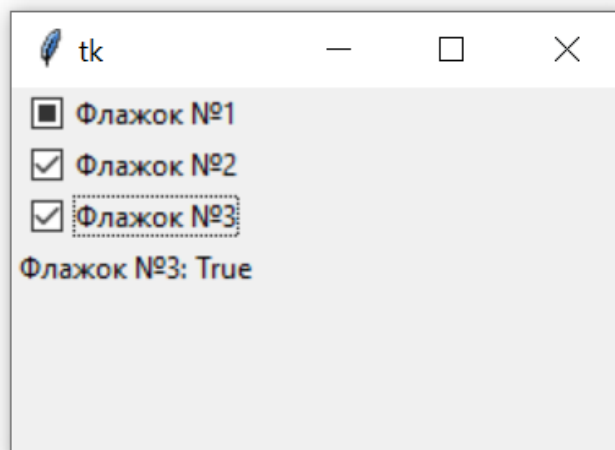
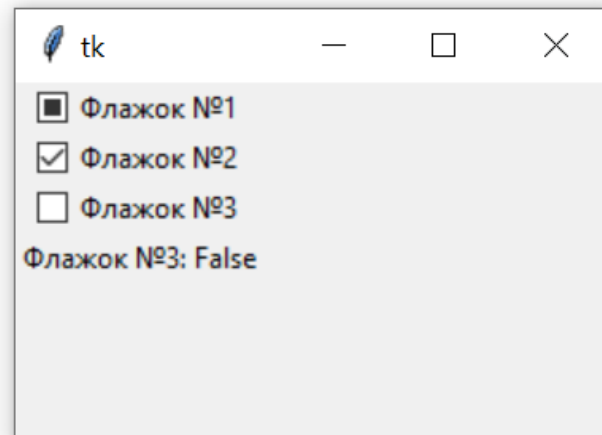
#создание переменной типа BooleanVar (из tkinter.ttk)
b2 = BooleanVar()
b2.set(True)
chk2 = Checkbutton(window, text='Флажок №2', var=b2)
chk2.grid(column=0, row=1)

def check():
    l1.config(text='Флажок №3: ' + str(b3.get()))

b3 = BooleanVar()
b3.set(0)
chk3 = Checkbutton(window, text='Флажок №3', var=b3, command = check)
chk3.grid(column=0, row=2)

l1 = Label(window, text='Флажок №3: ' + str(b3.get()))
l1.grid(column=0, row=4)

window.mainloop()
```



Создание селекторной кнопки (Radiobutton):

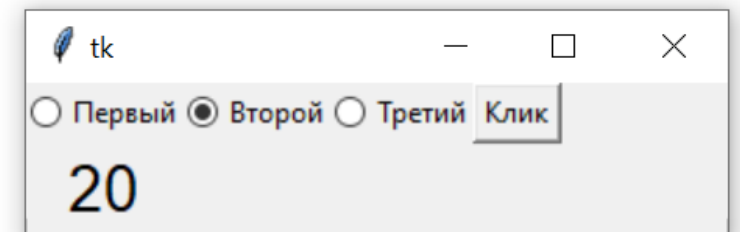
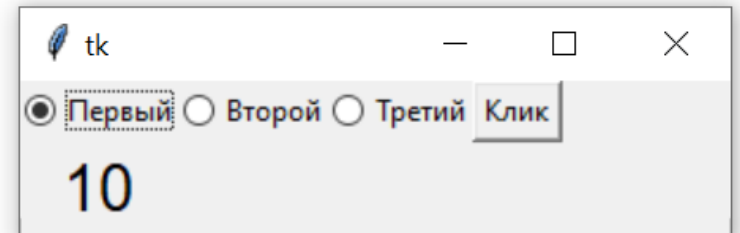
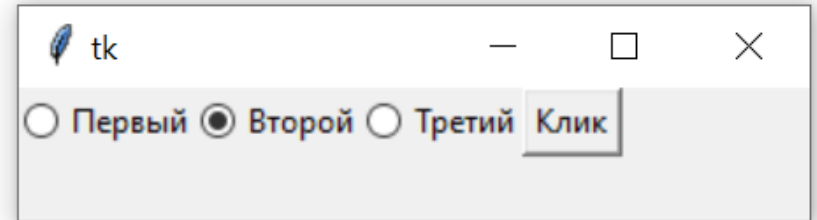
```
from tkinter import *  
from tkinter.ttk import Radiobutton  
  
def clicked():  
    lbl.configure(text=selected.get())
```

```
window = Tk()  
window.geometry('300x50')  
selected = IntVar()  
selected.set(20)
```

```
rad10 = Radiobutton(window, text='Первый', value=10, variable=selected)  
rad20 = Radiobutton(window, text='Второй', value=20, variable=selected)  
rad30 = Radiobutton(window, text='Третий', value=30, variable=selected)  
btn = Button(window, text="Клик", command=clicked)  
lbl = Label(window, font=("Arial Bold", 20))
```

```
rad10.grid(column=0, row=0)  
rad20.grid(column=1, row=0)  
rad30.grid(column=2, row=0)  
btn.grid(column=3, row=0)  
lbl.grid(column=0, row=1)
```

```
window.mainloop()
```

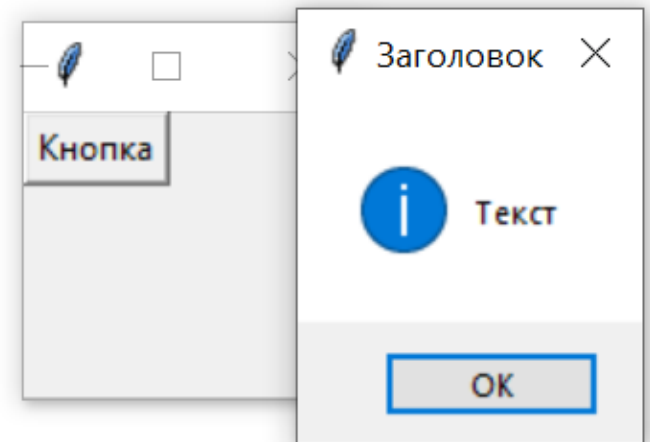


Всплывающие окна с сообщением:

```
from tkinter import *
from tkinter import messagebox

def clicked():
    messagebox.showinfo('Заголовок', 'Текст')

window = Tk()
window.geometry('100x100')
btn = Button(window, text='Кнопка', command=clicked)
btn.grid(column=0, row=0)
window.mainloop()
```



предупреждающее сообщение

```
messagebox.showwarning('Заголовок', 'Текст')
```

сообщение об ошибке

```
messagebox.showerror('Заголовок', 'Текст')
```

Выбор варианта:

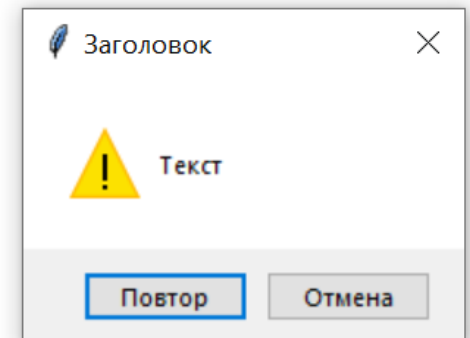
```
res = messagebox.askquestion('Заголовок', 'Текст')
```

```
res = messagebox.askyesno('Заголовок', 'Текст')
```

```
res = messagebox.askyesnocancel('Заголовок', 'Текст')
```

```
res = messagebox.askokcancel('Заголовок', 'Текст')
```

```
res = messagebox.askretrycancel('Заголовок', 'Текст')
```



Текстовая область (ScrolledText):

```
from tkinter import scrolledtext
txt = scrolledtext.ScrolledText(window,width=40,height=10)
txt.insert(...)
txt.delete(...)
```

Спинбокс (SpinBox):

```
spin = Spinbox(window, from_=0, to=100)
```

Диалог работы с файлами (FileDialog):

```
from tkinter import filedialog
file = filedialog.askopenfilename()
```

Создание меню:

```
from tkinter import *

window = Tk()
window.geometry('200x100')
menu = Menu(window)

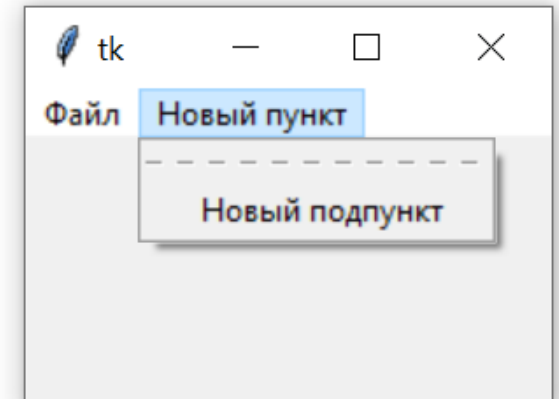
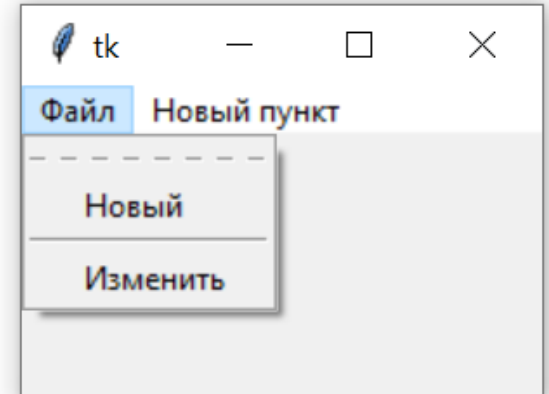
new_item = Menu(menu)
new_item.add_command(label='НОВЫЙ')
new_item.add_separator()
new_item.add_command(label='ИЗМЕНИТЬ')

menu.add_cascade(label='Файл', menu=new_item)

n_i = Menu(menu)
n_i.add_command(label='НОВЫЙ подпункт')
menu.add_cascade(label='НОВЫЙ пункт', menu=n_i)

window.config(menu=menu)

window.mainloop()
```



Фреймы (контейнеры):

```
from tkinter import *

window = Tk()
window.title("Главное окно")
window.minsize(width=250,height=250)

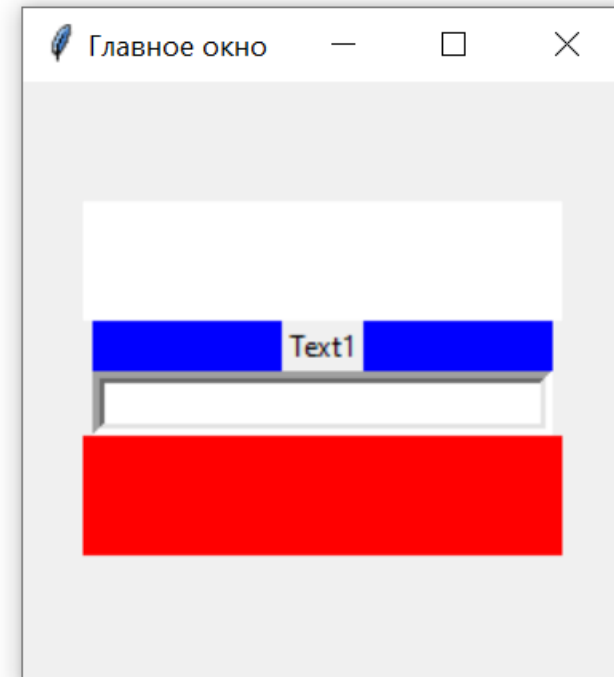
frame = Frame(window, width=200,height=50)
frame2 = Frame(window, width=200,height=50,bg="white")
frame3 = Frame(window, width=200,height=50,bg="blue")
frame4 = Frame(window, width=200,height=50,bg="red")

l1 = Label(frame3, text = 'Text1')
e1 = Entry(frame3, bd=5, width=30)

frame.pack()
frame2.pack()
frame3.pack()
frame4.pack()

l1.pack()
e1.pack()

window.mainloop()
```



Создание дочернего окна:

```
from tkinter import *  
  
def click():  
    children = Toplevel(window)  
    children.title("Дочернее окно")  
    children.minsize(width=200,height=200)  
  
window = Tk()  
window.geometry('150x150')  
btn1 = Button(window, text="OK", command = click)  
btn1.grid(column=0, row=0)  
window.mainloop()
```

