

Виджет *Button*

Этот виджет создает кнопку с текстом или изображением на ней. Создаем экземпляр объекта класса *Button*

```
объект_кнопка = Button(контейнер, параметры)
```

Параметр *text* задает текст для кнопки.

Параметры *width* и *height* задают соответственно ширину и высоту кнопки (единицей измерения являются символы).

Уже знакомые нам по другим виджетам параметры *bg* и *fg* задают соответственно цвет фона кнопки и цвет текста на ней.

Параметр *state* задает состояние кнопки – ее доступность или недоступность. Его значения *active* и *disabled* соответственно.

Например, создадим две кнопки – первая будет активна, вторая недоступна. Значение этого свойства по умолчанию *active*, поэтому для первой кнопки данный параметр можно было и не указывать.

```
btn1 = Button(tk, text="Кнопка1", state="active")
btn2 = Button(tk, text="Кнопка2", state="disabled")
```

Напомним, что менять параметры в процессе работы программы можно с помощью метода *config*. Например, можно первую кнопку сделать недоступной следующим образом:

```
btn1.config(state="disabled")
```

Важнейший параметр кнопки – параметр *command*. Его значение – имя функции, которая вызывается при нажатии на эту кнопку.

Функция – это подпрограмма, которая выполняет некоторые действия и возвращает какой-то результат. Обычно подпрограммы разделяют на процедуры и функции. Они отличаются только тем, что функция возвращает результат, а процедура – нет. Но в *Питоне* все подграммы определяются словом *def* и обычно называются функциями.

Если необходимо вернуть результат, то код функции должен содержать хотя бы один оператор *return результат*. Если же функция (как аналог процедуры) ничего не должна возвращать, а просто выполнять некоторые действия, то в ее коде может быть пустой оператор *return* или оператор *pass* или вообще ничего подобного не будет. Однако функция в *Питоне* всегда возвращает результат, даже если в ней нет *return* или присутствует *return* без возвращаемого значения. Тогда возвращаемый результат будет *None* — специальный тип данных, который дословно можно переводится как «ничего».

Приведем полный текст программы и разберем ее работу.

```
from tkinter import *
from PIL import Image, ImageTk
import pygame as pg

def m_play():
    snd.play()
    btn2.config(state="active")
    btn3.config(state="disabled")
    btn4.config(state="active")

def m_pause():
    pg.mixer.pause()
```

```

    btn2.config(state="disabled")
    btn3.config(state="active")

def m_unpause():
    pg.mixer.unpause()
    btn2.config(state="active")
    btn3.config(state="disabled")

def m_stop():
    snd.stop() # или pg.mixer.stop()
    btn2.config(state="disabled")
    btn3.config(state="disabled")
    btn4.config(state="disabled")

tk = Tk()
tk.title("Звук")
tk.geometry("700x500+0+0")
tk.resizable(width=False, height=False)

p = Image.open("music.jpg") # изображение размером 600x400
f = ImageTk.PhotoImage(p)
lbl = Label(tk, image=f)
lbl.image = f
lbl.place(x=50, y=5)

pg.mixer.init()
snd = pg.mixer.Sound("sound.wav")

btn1 = Button(tk, text="Воспроизвести", width=15,
command=m_play)
btn2 = Button(tk, text="Пауза", width=15, command=m_pause,
state="disabled")
btn3 = Button(tk, text="Продолжить", width=15,
command=m_unpause, state="disabled")
btn4 = Button(tk, text="Стоп", width=15, command=m_stop,
state="disabled")
btn1.place(x=50, y=450)
btn2.place(x=200, y=450)
btn3.place(x=350, y=450)
btn4.place(x=500, y=450)

tk.update()
tk.mainloop()
snd.stop() # или pg.mixer.stop()

```



Скриншот показывает первоначальное состояние окна – доступна только кнопка *Воспроизвести*.

При нажатии на нее вызывается функция *m_play*. В ней мы запускаем воспроизведение звука и меняем состояние остальных кнопок: оставляем кнопку *Воспроизвести* в прежнем активном состоянии (ведь пользователь в любой момент может захотеть прослушать звук сначала), делаем активными кнопки *Пауза* и *Стоп* (чтобы пользователь мог в любой момент приостановить или остановить совсем звук) и делаем недоступной кнопку *Продолжить* (ведь звук не остановлен, поэтому продолжать нечего).

Функция *m_pause* приостанавливает воспроизведение звука, оставляет для кнопок *Воспроизвести* и *Стоп* прежние активные состояния, делает кнопку *Продолжить* активной, а свою кнопку *Пауза* делает недоступной (ведь второй раз остановить остановленный звук уже не имеет смысла).

Функция *m_unpause* возобновляет воспроизведение звука, оставляет для кнопок *Воспроизвести* и *Стоп* прежние активные состояния, делает кнопку *Пауза* вновь активной, а свою кнопку *Продолжить* делает недоступной.

Функция *m_stop* останавливает воспроизведение звука, оставляет для кнопки *Воспроизвести* прежнее активное состояние и делает кнопки *Пауза*, *Продолжить* и *Стоп* недоступными (ведь остановленный звук можно только снова проиграть с начала).

Обратите ВНИМАНИЕ, что последней строкой кода программы является строка, останавливающая воспроизведение звука. Если ее не будет, то даже после закрытия окна звук будет воспроизводиться до конца.

Рассмотрим второй вариант программы. Он отличается только тем, что проигрывается звук формата MP3, и кнопки содержит не текст, а изображения формата PNG.



Полный текст программы:

```
from tkinter import *
from PIL import Image, ImageTk
import pygame as pg

def m_play():
    pg.mixer.music.play()
    btn2.config(state="active")
    btn3.config(state="disabled")
    btn4.config(state="active")

def m_pause():
    pg.mixer.music.pause()
    btn2.config(state="disabled")
    btn3.config(state="active")

def m_unpause():
    pg.mixer.music.unpause()
    btn2.config(state="active")
    btn3.config(state="disabled")

def m_stop():
    pg.mixer.music.stop()
    btn2.config(state="disabled")
    btn3.config(state="disabled")
```

```

    btn4.config(state="disabled")

tk = Tk()
tk.title("Звук")
tk.geometry("700x500+0+0")
tk.resizable(width=False, height=False)

p = Image.open("music.jpg") # изображение размером 600x400
f = ImageTk.PhotoImage(p)
lbl = Label(tk, image=f)
lbl.image = f
lbl.place(x=50, y=5)

pg.mixer.init()
pg.mixer.music.load("mauriat_bach.mp3")

img1=PhotoImage(file="play.png") # размеры кнопок 40x40
img2=PhotoImage(file="pause.png")
img3=PhotoImage(file="unpause.png")
img4=PhotoImage(file="stop.png")

btn1 = Button(tk, image=img1, width=60, height=60,
command=m_play)
btn2 = Button(tk, image=img2, width=60, height=60,
command=m_pause, state="disabled")
btn3 = Button(tk, image=img3, width=60, height=60,
command=m_unpause, state="disabled")
btn4 = Button(tk, image=img4, width=60, height=60,
command=m_stop, state="disabled")
btn1.place(x=50, y=420)
btn2.place(x=200, y=420)
btn3.place(x=350, y=420)
btn4.place(x=500, y=420)

tk.update()
tk.mainloop()
pg.mixer.music.stop()

```

Обратите внимание, что для кнопок задано изображение с помощью атрибута *image*. Изображение предварительно должно быть создано как экземпляр класса *PhotoImage* (вспомните миссию по работе с изображениями). Кроме того, в комбинации с параметром *image* для параметров *width* и *height* единицами измерения теперь стали пиксели, а не символы как в случае с текстом.

Еще одно отличие – использование методов модуля *mixer.music*, а не *mixer* (см. выше).