

GUI Help/Tkinter book

Материал из Викиучебника — открытых книг для открытого мира

[< GUI Help](#)

Содержание

- 1. Python
 - 1. [tkinter](#) Стандратный модуль для создания приложений с GUI интерфейсом.
- 2. VBA
 - 1. [VBA](#) GUI в среде MS Excel.
- 3. HTML
 - 1. [HTML book](#) Обзор GUI в HTML.
- 4. Pascal
 - 1. [Windows Forms](#) Интерфейс (API) для создания GUI-приложений.

Tkinter - библиотека для создания GUI-приложений на Python.

Содержание

- [Общий шаблон программ](#)
- [Основные классы](#)
 - [Классы](#)
 - [PhotoImage](#)
 - [Button](#)
 - [Radiobutton](#)
 - [Checkbutton](#)
 - [Entry](#)
 - [Text](#)
 - [Label](#)
 - [Scale](#)
 - [Scrollbar](#)
 - [Frame](#)
 - [LabelFrame](#)
 - [Listbox](#)
 - [Canvas](#)
 - [Фигуры](#)
 - [PanedWindow](#)
 - [Menu](#)
 - [Подменю](#)
 - [Виджеты в подменю](#)
- [Tk](#)
- [Курсор](#)
- [Toplevel](#)
- [Обработка событий](#)

<u>Упаковщики геометрии</u>
<u>Диалоговые окна</u>
<u>askopenfilename</u>
<u>asksavefilename</u>
<u>Работа с командной строкой</u>
<u>Примеры</u>

Общий шаблон программ

Все программы, написанные на Tkinter, должны иметь подобный шаблону вид:

```
from tkinter import *
root=Tk() # создал главное окно (всегда называйте его root)
#остальной код
root.mainloop() # теперь окно будет отображено при запуске
```

Основные классы

Модуль tkinter включает следующие классы:

- 1. **Button** (кнопка)
- 2. **Radiobutton** (радио-кнопка)
- 3. **Checkbutton** (флажок)
- 4. **Entry** (однострочное поле для ввода)
- 5. **Text** (многострочное поле для ввода)
- 6. **Label** (метка)
- 7. **Scale** (ползунок)
- 8. **Scrollbar** (полоса прокрутки)
- 9. **Frame** (виджет для группировки других виджетов)
- 10. **LabelFrame** (аналог Frame, только с заголовком)
- 11. **Listbox** (список)
- 12. **Canvas** (поле для рисования)
- 13. **PanedWindow** (элемент разделения окна)
- 14. **Menu** (главное меню)
- 15. **Tk** (главное единственное окно)
- 16. **Toplevel** (дочернее окно)

Классы

Ниже приведена схема, иллюстрирующая в какой части виджета работают какие свойства.



PhotoImage

PhotoImage - класс цветной картинки. Синтаксис создания изображения:

```
name = PhotoImage(file=p)
```

name - имя картинки (не имя файла загружаемого изображения), p - полный путь к изображению.

Button

Button - класс обычной кнопки. Общий синтаксис создания кнопки:

```
name = Button(window)
```

name - имя кнопки, window - имя окна, на котором она располагается.

Свойство	Значение
width	<p>Ширина кнопки.</p> <ul style="list-style-type: none"> Изначально равна такому количеству пикселей, чтобы текст помещался в кнопке вплотную к ее границам.
height	<p>Высота кнопки.</p> <ul style="list-style-type: none"> Изначально равна такому количеству пикселей, чтобы текст помещался в кнопке вплотную к ее границам.
text	<p>Текст на кнопке.</p> <ul style="list-style-type: none"> Изначально текст будет отображаться по центру кнопки. Возможно сделать многострочный текст, используя \n.
bg	<p>Фон кнопки.</p> <ul style="list-style-type: none"> Фон, который будет иметь кнопка в то время, когда на нее не нажали.
fg	<p>Цвет текста.</p> <ul style="list-style-type: none"> Цвет текста, который будет иметь кнопка в то время, когда на нее не нажали.
bd	<p>Ширина границы кнопки.</p> <ul style="list-style-type: none"> Ширина границы кнопки, которая будет у ее границы пока на нее не нажали.
activebackground	Цвет фона (когда кнопка нажата).
activeforeground	Цвет текста (когда кнопка нажата).
disabledbackground	Цвет фона (когда свойство state == DISABLED).
disabledforeground	Цвет текста (когда свойство state == DISABLED).
state	<p>Состояние кнопки (NORMAL, DISABLED).</p> <ul style="list-style-type: none"> NORMAL - обычное состояние кнопки, при котором она может нажиматься и взаимодействовать с пользователем. DISABLED - такое состояние кнопки, при котором она не может взаимодействовать с пользователем.
compound	<p>Расположение картинки на кнопке (CENTER, BOTTOM, LEFT, RIGHT, TOP).</p> <ul style="list-style-type: none"> Изначально картинка на кнопке будет отображаться вместо текста, но это можно изменить, изменяя значения свойства compound. BOTTOM - картинка будет отображаться под текстом. LEFT - картинка будет отображаться слева от текста. RIGHT - картинка будет отображаться справа от текста. TOP - картинка будет отображаться над текстом.

Свойство	Значение
width	Ширина радио-кнопки.
height	Высота радио-кнопки.
text	Текст на радио-кнопке.
bg	Фон радио-кнопки.
fg	Цвет текста.
bd	Ширина границы радио-кнопки.
activebackground	Цвет фона (когда радио-кнопка нажата).
activeforeground	Цвет текста (когда радио-кнопка нажата).
disabledbackground	Цвет фона (когда свойство state == DISABLED).
disabledforeground	Цвет текста (когда свойство state == DISABLED).
state	Состояние радио-кнопки (NORMAL, DISABLED).
compound	Расположение картинки на радио-кнопке (CENTER, BOTTOM, LEFT, RIGHT, TOP).
justify	Выравнивание текста.
relief	Рельеф радио-кнопки (FLAT, GROOVE, RIDGE, SUNKEN, RAISE).
overrelief	Рельеф радио-кнопки когда над ней находится курсор (FLAT, GROOVE, RIDGE, SUNKEN, RAISE).
image	Имя картинки, отображаемой на радио-кнопке.
selectimage	Имя картинки, отображаемой на радио-кнопке (когда она выбрана).
font	Вид шрифта на радио-кнопке.
indicatoron	Стиль отображения радио-кнопки (если true, то будет показываться кружок рядом с радио-кнопкой, иначе нет).
value	Значение, которое будет присваиваться переменной, указанной в параметре variable при выборе радио-кнопки.
variable	Имя переменной, у которой будет изменяться значение на указанное в свойстве value при выборе радио-кнопки.

Выбрать одну радио-кнопку можно только среди группы (радио-кнопок у которых значение параметра variable равны).

Checkbutton

Checkbutton - класс кнопки-флажка (флажка). Общий синтаксис создания кнопки:

```
name = Checkbutton(window)
```

name - имя кнопки-флажка, window - имя окна, на котором он располагается.

Свойство	Значение
width	Ширина флажка.
height	Высота флажка.
text	Текст около флажка.
bg	Фон флажка.
fg	Цвет флажка..
bd	Ширина границы флажка.
activebackground	Цвет фона (когда флажок нажат).
activeforeground	Цвет текста (когда флажок нажат).
disabledbackground	Цвет фона (когда свойство state == DISABLED).
disabledforeground	Цвет текста (когда свойство state == DISABLED).
state	Состояние флажка (NORMAL, DISABLED).
compound	Расположение картинки на флажке (CENTER, BOTTOM, LEFT, RIGHT, TOP).
justify	Выравнивание текста.
relief	Рельеф флажка (FLAT, GROOVE, RIDGE, SUNKEN, RAISE).
overrelief	Рельеф флажка когда над ним находится курсор (FLAT, GROOVE, RIDGE, SUNKEN, RAISE).
image	Имя картинки, отображаемой на кнопке-флажке.
selectimage	Имя картинки, отображаемой на кнопке-флажке (когда он выбран).
font	Вид шрифта на кнопке-флажке.
indicatoron	Стиль отображения флажка (если true, то будет показываться кружок рядом с ф, иначе нет).
onvalue	Значение, которое будет присваиваться переменной, указанной в параметре variable при выборе кнопки-флажка.
offvalue	Значение, которое будет присваиваться переменной, указанной в параметре variable при выборе другой (не данной) кнопки-флажка.
variable	Имя переменной, у которой будет изменяться значение на указанное в свойстве value при выборе кнопки-флажка.

Entry

Entry - класс однострочного текстового поля. Общий синтаксис создания однострочного текстового поля:

```
name = Entry(window)
```

name - имя однострочного текстового поля, window - имя окна, на котором оно располагается.

Свойство	Значение
width	Ширина поля.
bg	Фон поля.
fg	Цвет поля.
bd	Ширина границы поля.
activebackground	Цвет фона (когда в поле набирают текст).
activeforeground	Цвет текста (когда в поле набирают текст).
disabledbackground	Цвет фона (когда свойство state == DISABLED).
disabledforeground	Цвет текста (когда свойство state == DISABLED).
state	Состояние поля (NORMAL, DISABLED).
justify	Выравнивание текста.
highlightcolor	Цвет второй границы (когда поле имеет фокус).
highlightbackground	Цвет второй границы (когда поле не имеет фокус).
highlightthickness	Ширина второй границы.
relief	Рельеф текстового поля (FLAT, GROOVE, RIDGE, SUNKEN, RAISE).
overrelief	Рельеф текстового поля когда над ним находится курсор (FLAT, GROOVE, RIDGE, SUNKEN, RAISE).
font	Вид шрифта в однострочном текстовом поле.
textvariable	Имя переменной, в которой хранится весь текст, находящийся в поле.
selectbackground	Цвет фона выделенного фрагмента текста.
selectforeground	Цвет текста выделенного фрагмента текста.
insertontime	Время, которое курсор виден.
insertofftime	Время, которое курсор не видим.

Ниже а - имя текстового поля.

Функция	Значение
<code>a.get(начало,конец)</code>	Получает фрагмент текста от символа, позиция которого определяется первым параметром, до символа, позиция которого определена вторым параметром.
<code>a.insert(позиция,текст)</code>	Вставляет текст в поле перед символом, индекс которого указан в качестве параметра позиции. (Нумерация идет с 0.)
<code>a.delete(начало,конец)</code>	Удаляет текст от символа, позиция которого определяется первым параметром, до символа, позиция которого определена вторым параметром.

Text

Text - класс многострочного текстогого поля. Общий синтаксис создания многострочного текстового поля:

```
name = Text(window)
```

name - имя многострочного текстового поля, window - имя окна, на котором оно располагается.

Свойство	Значение
width	Ширина поля.
height	Высота поля.
bg	Фон поля.
fg	Цвет поля.
bd	Ширина границы поля.
activebackground	Цвет фона (когда в поле набирают текст).
activeforeground	Цвет текста (когда в поле набирают текст).
disabledbackground	Цвет фона (когда свойство state == DISABLED).
disabledforeground	Цвет текста (когда свойство state == DISABLED).
state	Состояние поля (NORMAL, DISABLED).
justify	Выравнивание текста.
highlightcolor	Цвет второй границы (когда поле имеет фокус).
highlightbackground	Цвет второй границы (когда поле не имеет фокус).
highlightthickness	Ширина второй границы.
relief	Рельеф текстового поля (FLAT, GROOVE, RIDGE, SUNKEN, RAISE).
overrelief	Рельеф текстового поля когда над ним находится курсор (FLAT, GROOVE, RIDGE, SUNKEN, RAISE).
font	Вид шрифта в многострочном текстовом поле.
selectbackground	Цвет фона выделенного фрагмента текста.
selectforeground	Цвет текста выделенного фрагмента текста.
insertontime	Время, которое курсор виден.
insertofftime	Время, которое курсор не видим.

Ниже а - имя текстового поля.

Функция	Значение
<code>a.get(начало,конец)</code>	<p>Получает фрагмент текста от символа, позиция которого определяется первым параметром, до символа, позиция которого определена вторым параметром. (Нумерация строк идет с 1, а символов в строке с 0.)</p> <ul style="list-style-type: none"> Позиция при указании должна иметь значение вида "%d.%d" % (x,y) Либо она может описываться одним и слов: CURRENT (текущая позиция курсора), END (позиция последнего символа в поле).
<code>a.insert(позиция,текст)</code>	<p>Вставляет текст в поле перед символом, индекс которого указан в качестве параметра позиции. (Нумерация строк идет с 1, а символов в строке с 0.)</p> <ul style="list-style-type: none"> Позиция при указании должна иметь значение вида "%d.%d" % (x,y) Либо она может описываться одним и слов: CURRENT (текущая позиция курсора), END (позиция последнего символа в поле).
<code>a.delete(начало,конец)</code>	<p>Удаляет текст от символа, позиция которого определяется первым параметром, до символа, позиция которого определена вторым параметром. (Нумерация строк идет с 1, а символов в строке с 0.)</p> <ul style="list-style-type: none"> Позиция при указании должна иметь значение вида "%d.%d" % (x,y) Либо она может описываться одним и слов: CURRENT (текущая позиция курсора), END (позиция последнего символа в поле).
<code>a.tag_add(имя, начало, конец)</code>	<p>Добавляет тег (объект, позволяющий форматировать текст) к полю а. (Нумерация строк идет с 1, а символов в строке с 0.)</p> <ul style="list-style-type: none"> Начало или конец должны иметь при указании значение вида "%d.%d" % (x,y) Либо она может описываться одним и слов: CURRENT (текущая позиция курсора), END (позиция последнего символа в поле).
<code>a.tag_configure(имя, параметры)</code>	<p>Изменяет свойства тега. Текст в диапазоне, который был указан при создании тега будет изменяться при смене свойств тега.</p> <ul style="list-style-type: none"> background - фон текста foreground - цвет текста

Label

Label - класс метки. Общий синтаксис создания метки:

```
name = Label(window)
```

name - имя метки, window - имя окна, на котором она располагается.

Свойство	Значение
width	Ширина метки.
height	Высота метки.
text	Текст на метке.
bg	Фон метки.
fg	Цвет текста.
bd	Ширина границы метки.
activebackground	Цвет фона (когда метка нажата).
activeforeground	Цвет текста (когда метка нажата).
disabledbackground	Цвет фона (когда свойство state == DISABLED).
disabledforeground	Цвет текста (когда свойство state == DISABLED).
state	Состояние метки (NORMAL, DISABLED).
compound	Расположение картинки на метке (CENTER, BOTTOM, LEFT, RIGHT, TOP).
justify	Выравнивание текста.
relief	Рельеф метки (FLAT, GROOVE, RIDGE, SUNKEN, RAISE).
overrelief	Рельеф метки когда над ней находится курсор (FLAT, GROOVE, RIDGE, SUNKEN, RAISE).
image	Имя картинки, отображаемой на метке.
font	Вид шрифта на метке.
textvariable	Имя переменной, в которой будет храниться текст на метке.

Scale

Scale - класс ползунка. Общий синтаксис создания ползунка:

```
name = Scale(window)
```

name - имя ползунка, window - имя окна, на котором он располагается.

Свойство	Значение
width	Ширина ползунка.
bg	Фон ползунка.
fg	Цвет текста рядом с ползунком.
bd	Ширина границы ползунка.
activebackground	Цвет фона (когда ползунок двигают или щелкают на него).
activeforeground	Цвет текста (когда ползунок двигают или щелкают на него).
disabledbackground	Цвет фона (когда свойство state == DISABLED).
disabledforeground	Цвет текста (когда свойство state == DISABLED).
state	Состояние ползунка (NORMAL, DISABLED).
justify	Выравнивание текста.
highlightcolor	Цвет второй границы (когда ползунок имеет фокус).
highlightbackground	Цвет второй границы (когда ползунок не имеет фокус).
highlightthickness	Ширина второй границы.
relief	Рельеф ползунка (FLAT, GROOVE, RIDGE, SUNKEN, RAISE).
overrelief	Рельеф ползунка когда над ним находится курсор (FLAT, GROOVE, RIDGE, SUNKEN, RAISE).
font	Вид шрифта у разметки ползунка.
from_	Первое число, отмечаемое рядом с ползунком.
to	Последнее число, отмечаемое рядом с ползунком.
tickinterval	Расстояние в пикселях через которое расставляются деления.
resolution	Минимальное число пикселей, на которое пользователь может передвинуть ползунок.
orient	Как ориентирована полоса с ползунком (HORIZONTAL, VERTICAL). <ul style="list-style-type: none">▪ HORIZONTAL - горизонтально▪ VERTICAL - вертикально
length	Длина линии, по которой движется ползунок.

Scrollbar

Scrollbar - класс полосы прокрутки. Общий синтаксис создания полосы прокрутки:

```
name = Scrollbar(window)
```

name - имя полосы прокрутки, window - имя окна, на котором она располагается.

Свойство	Значение
bg	Фон полосы прокрутки.
activebackground	Цвет фона (когда полосу прокрутки двигают).
disabledbackground	Цвет фона (когда свойство state == DISABLED).
state	Состояние поля (NORMAL, DISABLED).
relief	Рельеф полосы прокрутки (FLAT, GROOVE, RIDGE, SUNKEN, RAISE).
overrelief	Рельеф полосы прокрутки когда над ним находится курсор (FLAT, GROOVE, RIDGE, SUNKEN, RAISE).
command	<div>Имя виджета, содержание которого будет прокручиваться.</div> <div><ul style="list-style-type: none">Доступны только такие значения:<ul style="list-style-type: none">виджет.xview - виджет будет прокручиваться горизонтальновиджет.yview - виджет будет прокручиваться вертикальноДля возможности прокрутки виджета также нужно изменить прокручиваемый виджет:</div> <div>a.configure(nscrollcommand=s.set)</div> <div>н - либо х, либо у. Где а - прокручиваемый виджет, s - имя полосы прокрутки.</div>

Frame

Frame - класс фрейма. Общий синтаксис его создания:

```
name = Frame(window)
```

name - имя фрейма, window - имя окна, на котором он располагается.

Свойство	Значение
width	Ширина фрейма.
height	Высота фрейма.
bg	Фон фрейма.
activebackground	Цвет фона (когда полосу прокрутки двигают).
disabledbackground	Цвет фона (когда свойство state == DISABLED).
state	Состояние фрейма (NORMAL, DISABLED).
relief	Рельеф границы фрейма (FLAT, GROOVE, RIDGE, SUNKEN, RAISE).
highlightcolor	Цвет второй границы (когда фрейм имеет фокус).
highlightbackground	Цвет второй границы (когда фрейм не имеет фокус).
highlightthickness	Ширина второй границы.

LabelFrame

LabelFrame - класс фрейма с заголовком. Общий синтаксис его создания:

```
name = LabelFrame(window)
```

name - имя фрейма, window - имя окна, на котором он располагается.

Свойство	Значение
width	Ширина фрейма.
height	Высота фрейма.
bg	Фон фрейма.
activebackground	Цвет фона (когда полосу прокрутки двигают).
disabledbackground	Цвет фона (когда свойство state == DISABLED).
state	Состояние фрейма (NORMAL, DISABLED).
relief	Рельеф границы фрейма (FLAT, GROOVE, RIDGE, SUNKEN, RAISE).
highlightcolor	Цвет второй границы (когда фрейм имеет фокус).
highlightbackground	Цвет второй границы (когда фрейм не имеет фокус).
highlightthickness	Ширина второй границы.
text	Заголовок фрейма.

Listbox

Listbox - класс списка. Общий синтаксис его создания:

```
name = Listbox(window)
```

name - имя списка, window - имя окна, на котором он располагается.

Свойство	Значение
width	Ширина списка.
height	Высота списка.
bg	Фон списка.
disabledforeground	Цвет фона (когда свойство state == DISABLED).
state	Состояние списка (NORMAL, DISABLED).
relief	Рельеф границы списка (FLAT, GROOVE, RIDGE, SUNKEN, RAISE).
highlightcolor	Цвет второй границы (когда фрейм имеет фокус).
highlightbackground	Цвет второй границы (когда фрейм не имеет фокус).
highlightthickness	Ширина второй границы.
selectbackground	Цвет фона выделенных пунктов списка.
selectforeground	Цвет текста выделенных пунктов списка.
text	Заголовок списка.

Ниже a - имя списка.

Функция	Значение
a.get(начало,конец)	Получает фрагмент список из названий элементов списка. В список попадут элементы, индексы которых, находятся в диапазоне, определенным первым параметром и последним его параметром.
a.insert(позиция,название_элемента)	Вставляет элемент в список после данного индекса. Индекс указан в качестве параметра позиции. (Нумерация идет с 0.)
a.delete(начало,конец)	Удаляет из списка все элементы, индексы которых, попадают в диапазон, который начинается с индекса, определенным первым параметром и заканчивается индексом, определенным вторым параметром.

Canvas

Canvas - класс полотна для рисования. Общий синтаксис создания полотна:

```
name = Canvas(window)
```

name - имя полотна для рисования, window - имя окна, на котором оно располагается.

Свойство	Значение
width	Ширина полотна для рисования.
height	Высота полотна для рисования.
bd	Ширина границы полотна для рисования.
state	Состояние полотна для рисования (NORMAL, DISABLED).
highlightcolor	Цвет границы (когда на полотне рисуют).

Ниже а - окно полотно для рисования.

Функция	Описание
a.create_line(x,y,x1,y1)	Рисует линию между точками (x,y) и (x1,y1) на полотне а.
a.create_rectangle(x,y,x1,y1)	Рисует прямоугольник с верхней левой точкой (x,y) и нижней правой (x1,y1) на полотне а.
a.create_oval(x,y,x1,y1)	Рисует овал в границах прямоугольника с верхней левой точкой (x,y) и нижней правой (x1,y1) на полотне а.
a.create_polygon(x,y, ... ,xn,yn)	Рисует ломанную по точкам (x,y) ... (xn,yn) на полотне а.
a.create_text(x,y,x1,y1)	Рисует текст, центрируя его по точке (x,y) и (x1,y1), на полотне а.

Фигуры

Item - элемент (фигура), нарисованная на полотне Canvas. Синтаксис создания ссылки на item:

```
name = canvas_.method()
```

name - имя ссылки на item, canvas_ - имя Canvas, на котором рисуется фигура, method - одно из: create_line, create_rectangle, create_oval, create_polygon, create_text с соответственными методу параметрами.

PanedWindow

PanedWindow - класс разделителя окна. Общий синтаксис его создания:

```
name = PanedWindow(window,orient=k)
```

name - имя разделителя окна; window - имя окна, на котором он располагается; k - его ориентация (VERTICAL - вертикальна, HORIZONTAL - горизонтальна).

Свойство	Значение
bg	Фон элемента разделить окна.
sashpad	Толщина в линии разделителя.

Ниже а - элемент класса PanedWindow.

Функция	Описание
a.add(m)	Добавляет в элемент PanedWindow виджет m. Количество разделяющих виджет а линий становится равным k-1, где k - количество входящих в а виджетов.

Menu

Menu - класс главного меню. Общий синтаксис создания экземпляра меню:

```
name_main = Menu(window)
window.config(menu=name_main)
```

name_main - имя экземпляра меню, window - имя окна, на котором оно располагается.

Подменю

Для заполнения меню, используйте:

```
name_main.add_cascade(label=t,menu=name_undermenu)
```

name_main - имя кнопки меню, М - имя экземпляра меню, t - текст, являющийся отображаемым названием подменю.

Свойство	Значение
bg	Фон подменю.
fg	Цвет текста подменю.

Заполнять командами подменю можно так:

```
name_undermenu.add_command(label=text_,command=f)
```

name_undermenu - имя подменю, в которое добавиться команда; f - имя функции, которая выполнится при выбора данной команды.

Ниже а - подменю.

Функция	Описание
a.add_separator()	Вставляет горизонтальный разделитель после предыдущего пункта подменю а.
a.add_checkbutton()	Добавляет в подменю а кнопку-флажок.
a.add_radiobutton()	Добавляет в подменю а радио-кнопку..

Виджеты в подменю

Все виджеты подменю обладают основными свойствами виджетов вне меню.

Свойство	Значение
width	Ширина виджета.
height	Высота виджета.
text	Текст около виджета.
bg	Фон виджета.
fg	Цвет виджета.
bd	Ширина границы виджета.
activebackground	Цвет фона.
activeforeground	Цвет текста.
disabledbackground	Цвет фона (когда свойство state == DISABLED).
disabledforeground	Цвет текста (когда свойство state == DISABLED).
state	Состояние виджета (NORMAL, DISABLED).

Также у виджетов подменю присутствуют специфические для каждого класса виджетов свойства.

Tk

Tk - класс главной формы. Общий синтаксис создания окна:

```
name = Tk()
```

name - имя окна потомка.

Свойство	Значение
bg	Фон окна.

Ниже а - окно главное окно.

Функция	Описание
a.title(m)	Устанавливает заголовок m окна а.
a.minsize(x,y)	Устанавливает минимальный размер xху окна а.
a.maxsize(x,y)	Устанавливает максимальный размер xху окна а.

Курсор

Свойство	Значение
event.x	Позиция курсора по X во время события.
event.y	Позиция курсора по Y во время события.

Toplevel

Toplevel - класс окна верхнего уровня. Общий синтаксис создания окна:

```
name = Toplevel(window)
```

name - имя окна потомка, window - имя окна, потомком которого оно является.

Свойство	Значение
bg	Фон окна.

Ниже a - окно верхнего уровня.

Функция	Описание
a. title (m)	Устанавливает заголовок m окна a.
a. minsize (x,y)	Устанавливает минимальный размер x y окна a.
a. maxsize (x,y)	Устанавливает максимальный размер x y окна a.

Обработка событий

Для обработки событий пишите:

```
a.bind(e,f)
```

a - имя виджета, у которого настраивается реакция на событие e. При наступлении события будет выполниться функция f).

- Функция должна описываться следующим образом, если она «привязывается» к некоторому виджету методом bind:

```
def f(event):  
    #тело функции
```

- В случае, когда для этого используется метод command (функция будет срабатывать только про событии "<Button-1>"), пишется без event:

```
def f():  
    #тело функции
```

Общий вид описания события	Описание события
"<Button- m >"	<p>События нажатия одной из кнопок мыши (левой, правой и колесика).</p> <p>m должна быть равна одному из значений:</p> <ul style="list-style-type: none"> 1 - нажата левая кнопка мыши (курсор над виджетом) 2 - нажата правая кнопка мыши (курсор над виджетом) 3 - нажато колесико мыши (курсор над виджетом)
"< k >"	<p>Событие нажатия клавиши.</p> <p>k определяет название клавиши, которая нажата.</p>
"<B n -Motion>"	<p>Событие одновременного движения курсора мыши и нажатия на одну из кнопок мыши.</p> <p>n должна быть равна одному из значений:</p> <ul style="list-style-type: none"> 1 - нажата левая кнопка мыши во время движения курсора над виджетом 2 - нажата правая кнопка мыши во время движения курсора над виджетом 3 - нажато колесико мыши во время движения курсора над виджетом

Упаковщики геометрии

Упаковщик геометрии - функция, используемая 1 раз для того, чтобы виджет отобразился в нужном месте с настроенными параметрами на экране.

Ниже а - некоторый виджет.

Упаковщик	Описание
a.pack(side=g,fill=c)	<p>Отображает виджет ниже предыдущего если параметры не указаны, в противном случае если:</p> <ul style="list-style-type: none"> side==LEFT (состыковывает виджет к предыдущему к его левому краю или в плотную к левому краю окна) side==RIGHT (состыковывает виджет к правому левому краю или) side==TOP (состыковывает виджет к нижнему краю прошлого виджета либо верхнему краю окна) side==BOTTOM (состыковывает виджет к верхнему краю прошлого виджета либо нижнему краю окна) side==MIDDLE (помещает виджет в центр окна)
a.grid(row=c,column=m)	<p>Размещает виджет в колонке с координатами row (строка) и column (колонка).</p> <p>Размеры колонок и строк определяются размерами входящих в них виджетов.</p>
a.place(x=n,y=m)	<p>Размещает виджет так, что его <i>левый правый ???</i> угол будет в координатах, значений которых берутся из параметров x и y.</p>

Диалоговые окна

С помощью `tkinter.messagebox` можно создавать диалоговые окна. Для этого нужно отдельно его импортировать:

```
import tkinter.messagebox
```

вид диалогового окна	Описание
<code>messagebox.showinfo</code>	Создаёт окно с информацией
<code>messagebox.askyesno</code>	Создаёт окно по типу да-нет

Пример:

```
a=tkinter.messagebox.askyesno(message="Создаёт окно по типу да-нет")
```

askopenfilename

`askopenfilename` - класс окна открытия файла. Общий синтаксис создания окна:

```
name = askopenfilename()
```

`name` - имя окна.

После выборе файла `name` будет хранить путь к этому файлу, что позволит считать его содержимое.

asksavefilename

`asksavefilename` - класс окна сохранения файла. Общий синтаксис создания окна:

```
name = asksavefilename()
```

`name` - имя окна.

После выборе файла `name` будет хранить путь к этому файлу, что позволит сохранить в него содержимое.

Работа с командной строкой

Производится через модуль `os`:

```
import os
```

```
os.system('Команда')
```

Примеры

- Создание пустого окна с заголовком "Hello" и зелёным фоном:

```
from tkinter import *
root = Tk()
root.title('Hello')
root['bg'] = 'green'
root.mainloop()
```

Эта страница в последний раз была отредактирована 2 марта 2022 в 08:03.

Текст доступен по лицензии Creative Commons Attribution-ShareAlike, в отдельных случаях могут действовать дополнительные условия. Подробнее см. Условия использования.