

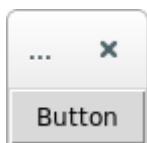
[back](#)

Przykłady użycia widgetów Tkinter

Przykłady pokazana poniżej zostały zaczerpnięte ze strony:

<https://github.com/Programmica/python-tkinter-examples>

Button



<sxh python> ' Showcasing a Button added to a Window, which when clicked, outputs a message to the user. '

```
import tkinter
```

```
class Button(tkinter.Tk):
```

```
    def __init__(self):  
        tkinter.Tk.__init__(self)  
        self.title("Button")
```

```
        button = tkinter.Button(text="Button", command=self.on_button_click)  
        button.pack(fill=tkinter.BOTH, expand=1)
```

```
    def on_button_click(self):  
        print("Button clicked")
```

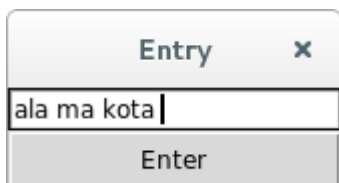
```
if name == "main":
```

```
    application = Button()  
    application.mainloop()
```

</sxh>



Entry



<sxh python> ' Entry example which takes any entered text and displays it when the user presses the button. '

```
import tkinter
```

```
class Entry(tkinter.Tk):
```

```
    def __init__(self):
        tkinter.Tk.__init__(self)
        self.title("Entry")
```

```
        self.entry = tkinter.Entry()
        self.entry.pack(fill=tkinter.BOTH, expand=0)
```

```
        button = tkinter.Button(text="Enter", command=self.on_button_click)
        button.pack(fill=tkinter.BOTH, expand=0)
```

```
    def on_button_click(self):
        print(self.entry.get())
```

```
if name == "main":
```

```
    application = Entry()
    application.mainloop()
```

</sxh>

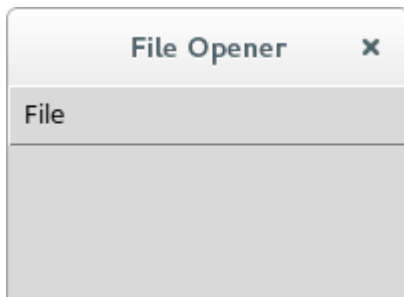


File dialogs

Wyczerpujący przykład można znaleźć na stronie: <https://tkinter.unpythonic.net/wiki/tkFileDialog>

Poniższy przykład został został zaczerpnięty z:

<https://gist.github.com/Yagisanatode/0d1baad4e3a871587ab1>



```
<sxh python> from tkinter import * from tkinter import ttk from
tkinter.filedialog import askopenfilename
```

```
root = Tk( )
```

```
#This is where we lauch the file manager bar. def OpenFile():
```

```

name =
askopenfilename(initialdir="C:/Users/Batman/Documents/Programming/tkinter/",
                 filetype =(("Text File", "*.txt"),("All
Files", "*..*")),
                 title = "Choose a file."
                 )

print (name)
#Using try in case user types in unknown file or closes without choosing a
file.
try:
    with open(name,'r') as UseFile:
        print(UseFile.read())
except:
    print("No file exists")

```

Title = root.title("File Opener")

#Menu Bar menu = Menu(root) root.config(menu=menu)

file = Menu(menu)

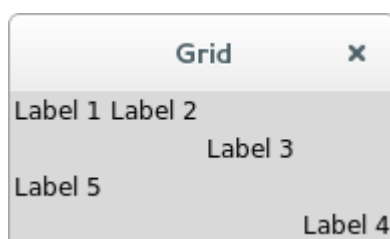
file.add_command(label = 'Open', command = OpenFile) file.add_command(label = 'Exit', command = lambda:exit())

menu.add_cascade(label = 'File', menu = file)

root.mainloop() </sxh>



Grid



<sxh python> ' The Grid container provides for multiple widgets to be inserted into a program based on defined rows and columns. '

import tkinter

class Grid(tkinter.Tk):

```

def __init__(self):
    tkinter.Tk.__init__(self)
    self.title("Grid")

```

```
grid = tkinter.Grid()
```

```
label = tkinter.Label(text="Label 1")
label.grid(row=0)
label = tkinter.Label(text="Label 2")
label.grid(row=0, column=1)
label = tkinter.Label(text="Label 3")
label.grid(row=1, column=2)
label = tkinter.Label(text="Label 4")
label.grid(row=4, column=3)
label = tkinter.Label(text="Label 5")
label.grid(row=2, column=0)
```

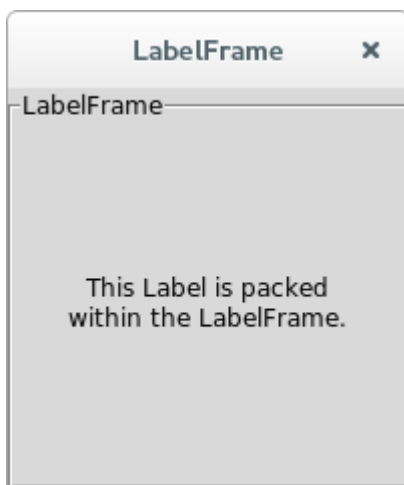
if `name == "main"`:

```
application = Grid()
application.mainloop()
```

</sxh>



LabelFrame



<sxh python> ' Showcasing a LabelFrame widget with a packed Label widget. The frame also contains 10 pixel padding providing a gap within the frame edge. '

```
import tkinter
```

```
class LabelFrame(tkinter.Tk):
```

```
    def __init__(self):
        tkinter.Tk.__init__(self)
        self.title("LabelFrame")
        self.geometry("200x200")
```

```
labelframe = tkinter.LabelFrame(text="LabelFrame", padx=10, pady=10)
```

```
labelframe.pack(fill=tkinter.BOTH, expand=1)
```

```
label = tkinter.Label(labelframe, text="This Label is packed\nwithin  
the LabelFrame")
```

e.")

```
label.pack(fill=tkinter.BOTH, expand=1)
```

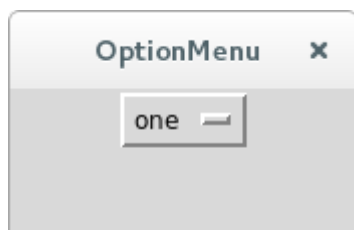
```
if name == "main":
```

```
    application = LabelFrame()  
    application.mainloop()
```

</sxh>



OptionMenu



<sxh python> ' The OptionMenu provides a dropdown from which an item can be selected from the pre-populated list. '

```
import tkinter
```

```
class OptionMenu(tkinter.Tk):
```

```
    def __init__(self):  
        tkinter.Tk.__init__(self)  
        self.title("OptionMenu")
```

```
        variable = tkinter.StringVar(self)  
        variable.set("two")
```

```
        optionmenu = tkinter.OptionMenu(self, variable, "one", "two", "three")  
        optionmenu.pack()
```

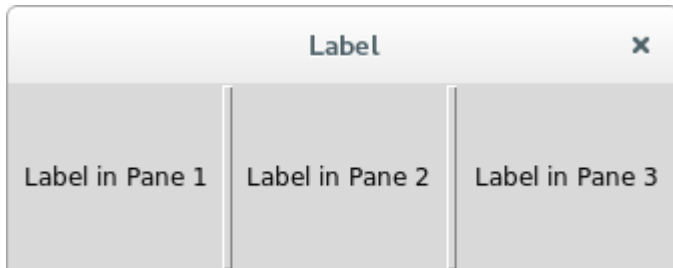
```
if name == "main":
```

```
    application = OptionMenu()  
    application.mainloop()
```

</sxh>



PanedWindow



<sxh python> ' An example of three labels packed into three separate PanedWindow areas with values set for the width of handle and sash. '

```
import tkinter
```

```
class PanedWindow(tkinter.Tk):
```

```
    def __init__(self):
        tkinter.Tk.__init__(self)
        self.title("Label")
```

```
        panedwindow = tkinter.PanedWindow()
        panedwindow.config(handlesize=10)
        panedwindow.config(sashwidth=5)
        panedwindow.config(sashrelief=tkinter.RAISED)
        panedwindow.pack(fill=tkinter.BOTH, expand=1)
```

```
        label = tkinter.Label(text="Label in Pane 1")
        panedwindow.add(label)
        label = tkinter.Label(text="Label in Pane 2")
        panedwindow.add(label)
        label = tkinter.Label(text="Label in Pane 3")
        panedwindow.add(label)
```

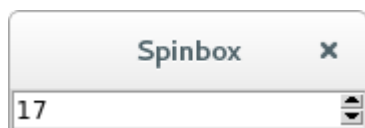
```
if name == "main":
```

```
    application = PanedWindow()
    application.mainloop()
```

</sxh>



Spinbox



<sxh python> ' Entry example which takes any entered text and displays it when the user presses the button. '

```
import tkinter
```

```
class Spinbox(tkinter.Tk):
```

```
    def __init__(self):  
        tkinter.Tk.__init__(self)  
        self.title("Spinbox")
```

```
        self.spinbox = tkinter.Spinbox(command=self.on_spinbox_change)  
        self.spinbox.config(from_=10, to=20)  
        self.spinbox.pack(fill=tkinter.BOTH, expand=0)
```

```
    def on_spinbox_change(self):  
        print(self.spinbox.get())
```

```
if name == "main":
```

```
    application = Spinbox()  
    application.mainloop()
```

</sxh>



[back](#)

From:
<https://www.l5.pk.edu.pl/~putanowr/dokuwiki/> - **Roman Putanowicz Wiki**

Permanent link:
https://www.l5.pk.edu.pl/~putanowr/dokuwiki/doku.php?id=pl:teaching:subjects:oop:lectures:gui:tkinter_widgets

Last update: **2017/10/02 15:37**

