

Zajęcia 12: GUI cz. 2

Python dla kognitywistów

Marcin Jukiewicz

Treść tych zajęć jest bardzo podobna do treści zajęć poprzednich. Większość programów, to programy z ostatnich zajęć przerobione tak, aby wykorzystywały klasy.

Ponadto zajmiemy się trzecim menadżerem układu "place()" oraz widżetem "checkboxbutton"

Puste okno

Poniższy kod otworzy puste okno o wymiarach 300 na 250 zatytułowane "Moje okno".

```
from Tkinter import *

class AplikacjaGUI(Frame, object):

    def __init__(self, master):
        super(AplikacjaGUI, self).__init__(master)
        self.master.title("Moje okno")
        self.master.geometry("300x250")

glowneOkno = Tk()
app = AplikacjaGUI(glowneOkno)
glowneOkno.mainloop()
```

Puste okno

Poniższy program także tworzy puste okno. Różnicą jest zastosowanie (dla uporządkowania kodu) funkcji `main()`.

```
from Tkinter import *

class AplikacjaGUI(Frame, object):

    def __init__(self, master):
        super(AplikacjaGUI, self).__init__(master)
        self.master.title("Moje okno")
        self.master.geometry("300x250")

def main():
    glowneOkno = Tk()
    app = AplikacjaGUI(glowneOkno)
    glowneOkno.mainloop()

main()
```

Okno z niedziałającym przyciskiem

```
from Tkinter import *  
  
class AplikacjaGUI(Frame, object):  
  
    def __init__(self, master):  
        super(AplikacjaGUI, self).__init__(master)  
        self.master.title("Moje okno")  
        self.master.geometry("300x250")  
        self.grid()  
        self.stworzWidgety()  
    def stworzWidgety(self):  
        przycisk1=Button(self, text = "Nic nie robie")  
        przycisk1.grid()  
  
def main():  
    glowneOkno = Tk()  
    app = AplikacjaGUI(glowneOkno)  
    glowneOkno.mainloop()  
main()
```

Okno z działającym przyciskiem i dodatkowym okienkiem

cz.1

```
from Tkinter import *
import tkMessageBox

class AplikacjaGUI(Frame, object):

    def __init__(self, master):
        super(AplikacjaGUI, self).__init__(master)
        self.master.title("Moje okno")
        self.master.geometry("300x250")
        self.grid()
        self.stworzWidgety()

    def stworzWidgety(self):
        przycisk1=Button(self, text = "Powitanie", comma
        przycisk1.grid()
```

Okno z działającym przyciskiem i dodatkowym okienkiem

cz.2

```
def akcjaPrzycisk(self):  
    tkMessageBox.showinfo("Okno powitalne", "Czesc")  
  
def main():  
    glowneOkno = Tk()  
    app = AplikacjaGUI(glowneOkno)  
    glowneOkno.mainloop()  
  
main()
```

Okno z działającym przyciskiem, dodatkowym okienkiem i polem tekstowym cz.1

```
from Tkinter import *
import tkMessageBox

class AplikacjaGUI(Frame, object):

    def __init__(self, master):
        super(AplikacjaGUI, self).__init__(master)
        self.master.title("Moje okno")
        self.master.geometry("300x250")
        self.pack()
        self.stworzWidgety()

    def stworzWidgety(self):
        ramka1 = Frame(self)
        ramka1.pack()
```


Okno z działającym przyciskiem, dodatkowym okienkiem i polem tekstowym cz.2

```
opisPolaTekstowego=Label(ramka1,
                           text="Wprowadz tekst")
opisPolaTekstowego.pack(side=LEFT)
poleTekstowe = Entry(ramka1)
poleTekstowe.pack(side=RIGHT)

ramka2 = Frame(self)
ramka2.pack()
przycisk1=Button(ramka2, text = "Powitanie",
                  command = self.akcjaPrzycisk)
przycisk1.pack(side=BOTTOM)

def akcjaPrzycisk(self):
    tkMessageBox.showinfo("Okno powitalne", "Czesc")
```

Okno z działającym przyciskiem, dodatkowym okienkiem i polem tekstowym cz.3

```
def main():  
    glowneOkno = Tk()  
    app = AplikacjaGUI(glowneOkno)  
    glowneOkno.mainloop()  
  
main()
```

Elementy w oknie należy ułożyć wg. pewnego określonego porządku, mogą posłużyć do tego trzy menadżery układu:

- pack(),
- grid(),
- place().

Do omówienia został nam "place()".

Zarządzanie układem: place()

```
from Tkinter import *

class AplikacjaGUI(Frame, object):

    def __init__(self, master):
        super(AplikacjaGUI, self).__init__(master)
        self.master.title("Moje okno")
        self.master.geometry("300x250")
        self.pack(fill=BOTH, expand=1)
        self.stworzWidzety()

    def stworzWidzety(self):
        przyciskCzerwony=Button(self, text="Czerwony",
                                fg="red")
        przyciskCzerwony.place(x=70, y=50)
        przyciskZielony=Button(self, text="Zielony",
                                fg="green")
        przyciskZielony.place(x=170, y=50)
```

Zarządzanie układem: place() cz.2

```
przyciskNiebieski=Button(self, text="Niebieski",  
fg="blue")  
przyciskNiebieski.place(x=70, y=150)  
przyciskBialy=Button(self, text="Bialy",  
fg="white")  
przyciskBialy.place(x=170, y=150)
```

```
def main():  
    glowneOkno = Tk()  
    app = AplikacjaGUI(glowneOkno)  
    glowneOkno.mainloop()
```

```
main()
```

Zarządzanie układem

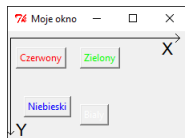
`pack()` pozwala na dokładanie kolejnych elementów wg. podanego kierunku.



`grid()` traktuje okno jako siatkę, należy wskazać dokładne współrzędne pola, w którym dany element ma zostać zamieszczony.



`place()` traktuje okno jako układ współrzędnych, należy wskazać dokładne współrzędne pola, w którym dany element ma zostać zamieszczony.



RadioButton z przyciskiem wywołującym

```
from Tkinter import *
import tkMessageBox

class AplikacjaGUI(Frame, object):

    def __init__(self, master):
        super(AplikacjaGUI, self).__init__(master)
        self.master.title("Moje okno")
        self.master.geometry("300x250")
        self.pack()
        self.stworzWidgety()

    def stworzWidgety(self):
        self.wartosc = IntVar()
        rprzycisk1 = Radiobutton(self, text="Wybor 1",
                                  variable=self.wartosc,
                                  value=1,
                                  command=self.akcjaPrzycisk)
```

RadioButton z przyciskiem wywołującym cz.2

```
rprzycisk1.pack()
rprzycisk2 = Radiobutton(self, text="Wybor 2",
                          variable=self.wartosc,
                          value=2,
                          command=self.akcjaPrzycisk)
rprzycisk2.pack()
```

```
def akcjaPrzycisk(self):
    if self.wartosc.get()==1:
        tkMessageBox.showinfo("Say Hello", "Witaj")
    if self.wartosc.get()==2:
        tkMessageBox.showinfo("Say Hello", "Czesc")
```

```
def main():
    glowneOkno = Tk()
    app = AplikacjaGUI(glowneOkno)
    glowneOkno.mainloop()

main()
```


CheckButton cz.1

```
from Tkinter import *

class AplikacjaGUI(Frame, object):

    def __init__(self, master):
        super(AplikacjaGUI, self).__init__(master)
        self.master.title("Moje okno")
        self.master.geometry("300x250")
        self.pack()
        self.stworzWidgety()

    def stworzWidgety(self):
        self.wartosc = IntVar()
        cprzycisk = Checkbutton(
            self,
            text="Pokaz tytul okna",
            variable=self.wartosc,
            command=self.akcjaPrzycisk)
```

CheckBox cz.2

```
cprzycisk.place(x=50, y=50)
```

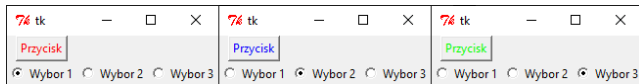
```
def akcjaPrzycisk(self):  
    if self.wartosc.get()==1:  
        self.master.title("Checkbutton")  
    else:  
        self.master.title("")
```

```
def main():  
    glowneOkno = Tk()  
    app = AplikacjaGUI(glowneOkno)  
    glowneOkno.mainloop()
```

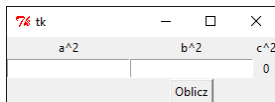
```
main()
```

Zadania

1. Przygotuj okno z trzema przyciskami. Niech każdy z nich otwiera inne okno (tkMessageBox).
2. Przygotuj okno, w którym RadioButtony pozwolą na zmianę koloru czcionki przycisku.



- a) użyj pack(),
 - b) użyj grid(),
 - c) użyj place().
3. Przygotuj okno, w którym 3 przyciskami sterujemy tekstem wyświetlanym w etykiecie.
 4. Przygotuj program do obliczeń związanych z twierdzeniem Pitagorasa.



5. Przygotuj kalkulator.
6. Przygotuj program, który zlicza wciśnięcia przycisku.