



GRAFIČKO KORISNIČKO SUČELJE

KRATAK UVOD

- većinom smo do sada koristili interaktivno sučelje ili datoteke za rad s programima (praktično kad želimo nešto isprobati)
- želimo li da se naša aplikacija koristi moramo je prilagoditi korisnicima (pogotovo sučelje)
- za kreiranje grafičkog sučelja koristimo modul tkinter (dio osnovnog Pythonovog paketa)
- osnovni element u grafičkom sučelju je – prozor (Tk)
- u prozor dodajemo ostale elemente npr. tekstualne okvire, polja za unos, gumbe... (ovo su sve posebne klase unutar modula tkinter)

PRVI PROZOR – KLASA TK

- `p = Tk()` – kreira početni prozor `p` te onda na njemu možemo koristiti metode i mijenjati neke attribute na kraju završavamo s `mainloop()`
- neke od metoda:
 - `title(Naslov)`
 - `config()` – mijenja attribute prozora

naziv metode	opis
<code>background, bg</code>	pozadinska boja
<code>borderwidth, bd</code>	debljina ruba prozora
<code>height</code>	visina prozora
<code>padx, pady</code>	udaljenost sadržaja od rubova prozora
<code>width</code>	širina prozora

- `destroy()` – zatvara prozor

PR. 1.

Kreirajte prozor dimenzija 200*300. Stavite mu naslov „Prvi prozor” te boju pozadine stavite na neku zelenu.

RJEŠENJE PR. 1.

```
from tkinter import* #dodamo mogul koji ćemo koristiti tj. tkinter

prozor = Tk() #kreira prozor
prozor.config(height = 200, width = 300, background = "#00AA00") #zadana svojstva
prozor.title("Prvi prozor")
prozor.mainloop()#mora biti na kraju
```

DODATNO (PROŠLA GODINA) – DRUGI NAČIN

- za veličinu možemo koristiti i geometry (položaj i veličina prozora) te metodu resizable za promjenu veličine prozora

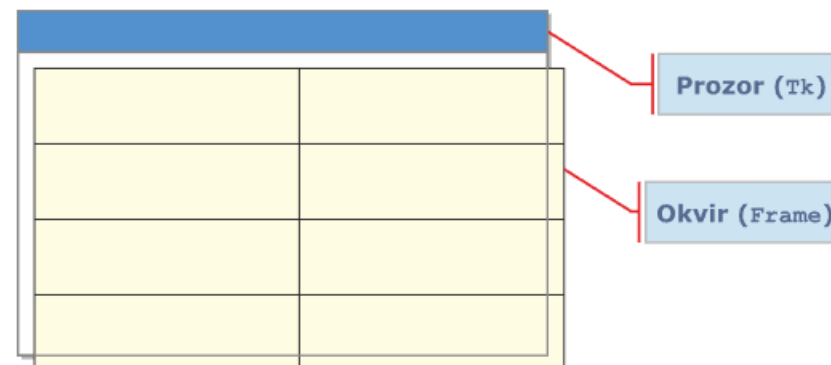
```
from tkinter import*
```

```
prozor = Tk()  
prozor.geometry("300x200+100+100") #veličina prozora i lokacija na zaslonu  
prozor.config(background = "green") #boja  
prozor.resizable(True, True) #možemo mijenjati širinu prozora  
prozor.title("Prvi prozor")  
prozor.mainloop() #mora biti na kraju
```


OKVIR – KLASA FRAME

- okvir (engl. Frame) - služi za organiziranje elemenata u sučelju
- veličina prozora inicijalno se prilagođava veličini okvira.
- na okvir stavljamo ostale elemente sučelja na jedan od sljedećih načina:

upravitelj	metoda	opis
<i>Grid</i>	<code>grid()</code>	okvir se podijeli u retke i stupce te se elementi sučelja stavljaju u odgovarajuće ćelije pripremljenog rastera
<i>Pack</i>	<code>pack()</code>	elementi sučelja dodaju se na okvir jedan iza drugoga – prema vrhu okvira, prema dnu okvira, ulijevo ili udesno (<i>TOP, BOTTOM, LEFT, RIGHT</i>)
<i>Place</i>	<code>place()</code>	svaki element stavljamo na točno određeno mjesto (točno određene koordinate)



PR. 2.

Kreirajmo prozor veličine 600×400 s plavom bojom pozadine te na njemu okvir veličine 300×200 s crvenom bojom pozadine i gridom od četiri retka i tri stupca.

RJEŠENJE PR. 2.

```
from tkinter import*
```

```
prozor = Tk()
```

```
prozor.config(height = 600, width = 400, bg = "blue")
```

```
f = Frame(prozor, height = 300, width = 200, bg = "red")
```

```
f.grid(rows = 4, columns = 3)
```

```
prozor.mainloop()
```

GUMB

- gumb ćemo kreirati klasom Button
- pokrećemo neku aktivnost na prozoru

upravitelj	opis									
<code>text</code>	tekst koji se nalazi na gumbu									
<code>width</code>	širina gumba (broj znakova koji mogu stati na gumb)									
<code>height</code>	visina gumba (broj redaka znakova koji mogu stati na gumb)									
<code>bg, background</code>	pozadinska boja									
<code>fg, foreground</code>	boja teksta									
<code>font</code>	font kojim će biti ispisan sadržaj gumba: font se zadaje trojkom, pri čemu su elementi trojke: <ul style="list-style-type: none">▪ naziv fonta – <i>Arial, Verdana...</i>▪ veličina fonta▪ stil – <i>bold, normal, italic</i>, a može se i kombinirati <i>bold italic</i>									
<code>anchor</code>	pozicija teksta na gumbu, definira se geografskim oznakama <i>N</i> – gore, <i>S</i> – dolje, <i>E</i> – desno, <i>W</i> – lijevo, <i>CENTER</i> – sredina, a moguće su i kombinacije kao primjerice <i>NE</i> (gore desno) <table><tbody><tr><td><i>NW</i></td><td><i>N</i></td><td><i>NE</i></td></tr><tr><td><i>W</i></td><td><i>CENTER</i></td><td><i>E</i></td></tr><tr><td><i>SW</i></td><td><i>S</i></td><td><i>SE</i></td></tr></tbody></table>	<i>NW</i>	<i>N</i>	<i>NE</i>	<i>W</i>	<i>CENTER</i>	<i>E</i>	<i>SW</i>	<i>S</i>	<i>SE</i>
<i>NW</i>	<i>N</i>	<i>NE</i>								
<i>W</i>	<i>CENTER</i>	<i>E</i>								
<i>SW</i>	<i>S</i>	<i>SE</i>								

<code>bd, borderwidth</code>	debljina ruba
<code>command</code>	naredba koja se izvodi klikom na gumb
<code>textvariable</code>	naziv varijable s pomoću koje se kontrolira tekst koji piše na gumbu
<code>image</code>	slika na gumbu
<code>justify</code>	kako će biti poravnan tekst ako je napisan u više redaka: <i>LEFT, CENTER, RIGHT</i>
<code>padx</code>	udaljenost teksta od lijevog/desnog ruba gumba
<code>pady</code>	udaljenost teksta od gornjeg/donjeg ruba gumba
<code>state</code>	je li gumb aktivan – <i>NORMAL</i> ili je neaktivan – <i>DISABLED</i>

PARAMETRI METODE `grid()`

naziv parametra	opis
<code>row</code>	red unutar tablice u kojem će se nalaziti gumb
<code>column</code>	stupac unutar tablice u kojem će se nalaziti gumb
<code>rowspan</code>	broj redaka kroz koje se proteže ćelija u kojoj će se nalaziti gumb
<code>columnspan</code>	broj stupaca kroz koje se proteže ćelija u kojoj će se nalaziti gumb
<code>padx</code>	udaljenost ruba gumba od linije tablice – lijevo i desno
<code>pady</code>	udaljenost ruba gumba od linije tablice – gore i dolje
<code>sticky</code>	položaj gumba unutar ćelije tablice, vrijednosti su iste kao za parametar <code>anchor</code> (vidi tablicu 2.3)

ZADATAK 1.

Rasporedite gumbe kao na slici ako su oni smješteni u crvenom okviru koji je podijeljen na 3 stupca i 3 retka:



NALJEPNICA

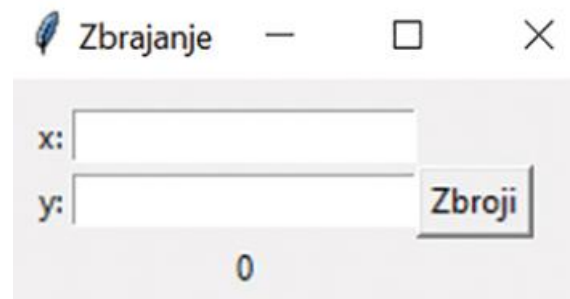
- služi za ispis teksta ili postavljanje slike – klasa Label
- parametri koje možemo koristiti su slični kao i kod gumba

UNOS

- služi za unos teksta s tipkovnice kroz grafičko sučelje – klasa Entry
- parametri koje možemo koristiti su slični kao i kod naljepnice i gumba
- `get()`, `insert(indeks, s)`, `delete(prvi, zadnji)`

PR. 3.

Kerirajmo ovakvo sučelje kakvo je na slici te dodajmo na gumb funkciju koja će uzeti unesene brojeve i zbrojiti ih te umjesto nule upisati zbroj.



The image shows a small application window titled "Zbrajanje" (Addition). It features two input fields labeled "x:" and "y:". To the right of the "y:" field is a button labeled "Zbroji" (Add). Below the input fields, the number "0" is displayed, representing the current result of the addition.

RJEŠENJE PR. 3.

```
from tkinter import*

def zbroji():
    c = int(U1.get())+int(U2.get())
    T3.config(text = c)

prozor = Tk()
prozor.title("Zbroji")
okvir = Frame(prozor)
okvir.grid(rows = 3, columns = 3, padx = 10, pady = 10)
T1 = Label(okvir, text = "x:")
T1.grid(row = 1, column = 1)
U1 = Entry(okvir)
U1.grid(row = 1, column = 2)
T2 = Label(okvir, text = "y:")
T2.grid(row = 2, column = 1)
U2 = Entry(okvir)
U2.grid(row = 2, column = 2)
G = Button(okvir, text = "Zbroji", command = lambda:zbroji())
G.grid(row = 2, column = 3)
T3 = Label(okvir, text = "0")
T3.grid(row = 3, column = 2)
prozor.mainloop()
```

OPCIONALNO (StringVar)

- olakšavaju dohvaćanje i mijenjanje sadržaja elemenata sučelja
- promjenom vrijednosti StringVar varijable u nekoj metodi, promijenit će se sadržaj svih elemenata sučelja koji koriste tu varijablu
- metode `get()` i `set(v)`
- moramo ih povezati s elementom sučelja koristeći `textvariable`

RJEŠENJE PR. 3. – DRUGI NAČIN

```
from tkinter import*

def zbroji():
    v3.set(int(v1.get())+int(v2.get()))

prozor = Tk()
prozor.title("Zbroji")
okvir = Frame(prozor)
okvir.grid(rows = 3, columns = 3, padx = 10, pady = 10)
T1 = Label(okvir, text = "x:")
T1.grid(row = 1, column = 1)
v1 = StringVar()
U1 = Entry(okvir, textvariable = v1)
U1.grid(row = 1, column = 2)
T2 = Label(okvir, text = "y:")
T2.grid(row = 2, column = 1)
v2 = StringVar()
U2 = Entry(okvir, textvariable = v2)
U2.grid(row = 2, column = 2)
G = Button(okvir, text = "Zbroji", command = lambda:zbroji())
G.grid(row = 2, column = 3)
v3 = StringVar()
v3.set(0)
T3 = Label(okvir, textvariable = v3)
T3.grid(row = 3, column = 2)
prozor.mainloop()
```

ZADATAK 2.

Kreirajte grafičko sučelje što sličnije ovom na slici i da gumbići (A+B, A-B, A*B i A/B) računaju određene operacije.

A:		B:	
0		0	
A+B	A-B	A*B	A/B
0			

DODATNI ELEMENTI GRAFIČKOG SUČELJA

- lista za odabir opcija (Listbox)
- gumb isključivog izbora (Radiobutton)
- padajuća lista (OptionMenu)
- Gumbi višestrukog izbora (Checkbutton)

RADIOBUTTON

- želimo samo jednu opciju

```
from tkinter import*

prozor = Tk()
prozor.title("Zbroji")
okvir = Frame(prozor)
okvir.grid(rows = 3, columns = 3, padx = 10, pady = 10)
Vrste = ["Obična", "Crna", "Integralna"]
v = IntVar()
v.set(0)
L = []
for i in range(3):
    L.append(Radiobutton(okvir, text = Vrste[i], variable = v, value = i))
    L[i].grid(row = i+1, column = 1)
prozor.mainloop()
```


CHECKBUTTON

- omogućuje nam višestruki odabir

```
from tkinter import*

prozor = Tk()
okvir = Frame(prozor)
okvir.grid(rows = 3, columns = 3)
L = ["Obicna", "Crna", "Integralna"]
LV = []
CB = []
for i in range(3):
    LV.append(BooleanVar())
    CB.append(Checkbutton(okvir, text=L[i], variable=LV[i]))
    CB[i].grid(row=i + 1, column=1, sticky=W)
prozor.mainloop()
```

OPTIONMENU

- padajući izbornik

```
from tkinter import*
```

```
prozor = Tk()  
okvir = Frame(prozor)  
okvir.grid(rows = 3, columns = 3)  
v = StringVar()  
v.set("Obicna")  
OptionMenu(okvir, v, "Obicna", "Crna", "Integralna").grid(row=1, column = 1)  
prozor.mainloop()
```

LISTBOX

naziv parametra	opis
<code>width</code>	širina liste (broj znakova)
<code>height</code>	visina liste (broj redaka – Inicijalno 10)
<code>bg, background</code>	pozadinska boja
<code>fg, foreground</code>	boja teksta
<code>font</code>	font kojim će biti ispisane stavke liste
<code>bd, borderwidth</code>	debljina ruba
<code>highlightbackground</code>	pozadinska boja dok je lista aktivna
<code>highlightcolor</code>	boja teksta dok je lista aktivna
<code>selectbackground</code>	pozadinska boja odabrane stavke
<code>selectforeground</code>	boja teksta odabrane stavke
<code>selectmode</code>	<p>način odabira stavki:</p> <ul style="list-style-type: none">▪ <i>BROWSE</i> – ovo je inicijalna opcija, moguće je odabrati samo jednu stavku, ali dok je jedna stavka odabrana povlačenjem miša, držeći lijevi gumb miša pritisnut odabrat će se i druge stavke▪ <i>SINGLE</i> – moguće je odabrati samo jednu stavku, nije moguće odabrati druge stavke povlačenjem miša dok se drži pritisnut lijevi gumb miša▪ <i>MULTIPLE</i> – moguće je odabrati više stavki klikajući redom na stavke koje želimo odabrati▪ <i>EXTENDED</i> – moguće je odabrati bilo koji niz susjednih stavki na način da se odabere prva stavka te se pritisne tipka (<i>Shift</i>), a potom se klikne na posljednju stavku koju želimo odabrati; isto je moguće i na način da kliknemo na prvu stavku koju želimo odabrati te držeći pritisnut lijevi gumb miša, vučemo do posljednje stavke koju želimo odabrati

naziv metode	opis
<code>curselection()</code>	vraća listu indeksa svih odabranih linija u listi
<code>delete(p, k=None)</code>	iz liste briše sve stavke od pozicije p do pozicije k (uključujući obje pozicije), ako parametar k nije definiran, briše samo stavku s indeksom p
<code>get(p, k=None)</code>	vraća n -torku stavki počevši od stavke s indeksom p do stavke s indeksom k (uključivo), pri čemu ako stavka k nije postavljena, uzimat će samo stavku s indeksom p
<code>insert(i, e)</code>	u listu dodaje stavku e ispred elementa čiji je indeks i ; ako stavku želimo dodati na kraj liste kao vrijednost parametra i , koristit ćemo konstantu <i>END</i>
<code>selection_clear(p, k=None)</code>	odznačit će odabir svih stavki od stavke s indeksom p do stavke s indeksom k (uključivo), ako k nije definiran, odznačit će samo stavku s indeksom p
<code>selection_includes(i)</code>	vraća 1 ako je stavka s indeksom i selektirana, a inače vraća 0
<code>selection_set(p, k=None)</code>	odabire sve stavke od stavke s indeksom p do stavke s indeksom k (uključivo), ako k nije postavljen, odabire samo stavku p