Uživatelská rozhraní cvičení 3

Na dnešním cvičení si představíme druhý způsob, jak v okně, případně v kontejneru (typicky Frame nebo LabelFrame), rozložit jednotlivé prvky. Jedná se o funkci **grid**, která umožňuje rozložit prvky do mřížky. Povinnými parametry jsou **row** a **column**, kterými určíme buňku mřížky, kde se má prvek vykreslit. Znázorníme si to stejně jako minule na tlačítkách.

```
b1 = Button(root, text="Button1", bg="yellow")
b2 = Button(root, text="Button2", bg="red")
b3 = Button(root, text="Button3", bg="green")

b1.grid(row=0, column=0)
b2.grid(row=0, column=1)
b3.grid(row=1, column=0)
```

Indexování sloupců nemusí začínat od 0, řádky nebo sloupce, ve kterých nejsou žádné komponenty budou skryty. Můžeme také slučovat některé buňky dohromady a do výsledné oblasti pak vkládat jednotlivé prvky. Použijeme k tomu parametry rowspan a columnspan, kdy tuto hodnotu nastavíme na počet buňek v řádku nebo sloupci. Index row/column nastavíme na ten, kde má komponenta začínat. Pokud tedy chceme při vkládání tlačítka b3 sloučit buňky v prvním řádku přes dva sloupce, nastavíme vykreslení takto:

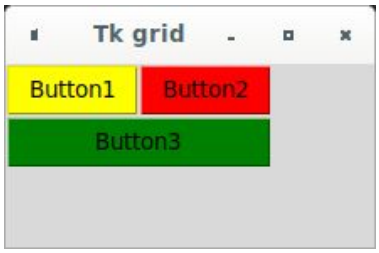
```
b1.grid(row=0, column=0)
b2.grid(row=0, column=1)
b3.grid(row=1, column=0, columnspan=2)
```

Podobně jako u příkazu pack můžeme komponentu uvnitř zarovnávat, pokud je kolem ní nějaké místo a k tomu použijeme parametr **sticky**, který lze nastavit na hodnoty W, E, N, S a jejich kombinace. Pokud tedy chceme prvek pouze zarovnat na pravou stranu, použijeme **sticky=E**, pokud jej chceme horizontálně roztáhnout, použijeme **sticky=W+E.**

```
b1.grid(row=0, column=0)
b2.grid(row=0, column=1)
b3.grid(row=1, column=0, columnspan=2, sticky=E)
```

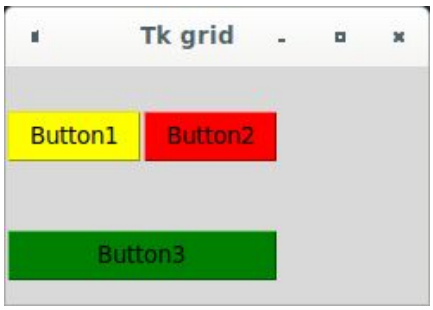

```
b1.grid(row=0, column=0)
b2.grid(row=0, column=1)
b3.grid(row=1, column=0, columnspan=2, sticky=E+W)
```

Mřížka i s komponenty se defaultně zobrazuje v minimálním stavu, takže po zvětšení okna obsah nevyplní.



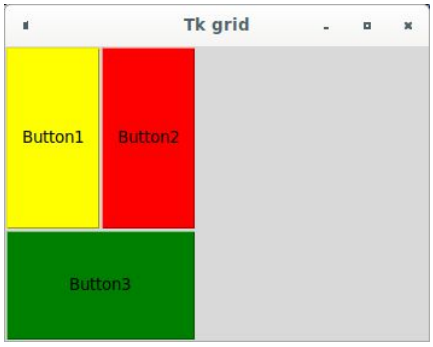
Pokud chceme, nastavit některý řádek nebo sloupec, aby se zvětšoval, nastavení pomocí **rowconfigure** nebo **columnconfigure**. Tyto metody mají dva parametry - id řádku/sloupce, který má reagovat na změnu velikosti okna (lze to tedy nastavit pouze u některých) a váhu, kterou říkáme, jak se má prostor vyplňovat vůči ostatním (pokud má jeden sloupec váhu 2, druhý sloupec váhu 1, budou se zvětšovat v poměru 2:1).

```
b1.grid(row=0, column=0)
b2.grid(row=0, column=1)
b3.grid(row=1, column=0, columnspan=2, sticky=W+E)

root.rowconfigure(0, weight=2)
root.rowconfigure(1, weight=1)
```



V tuto chvíli mají komponenty přiděleny více prostoru,
podobně jako v případě **expand** u **pack**. Lze je teď tedy roztáhnout i do dalších stran.

```
b1.grid(row=0, column=0, sticky=N+S)
b2.grid(row=0, column=1, sticky=N+S)
b3.grid(row=1, column=0, columnspan=2, sticky=W+E+N+S)

root.rowconfigure(0, weight=2)
root.rowconfigure(1, weight=1)
```



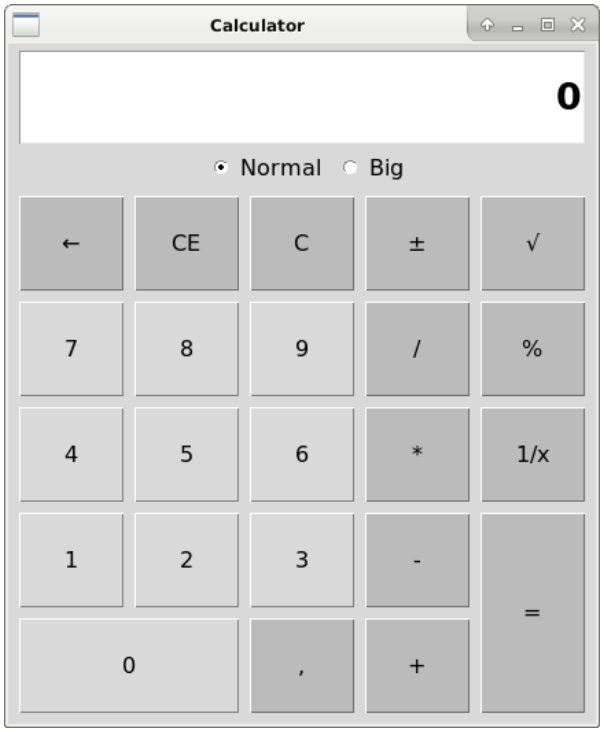
Stejně jako u funkce **pack**, můžete u **grid** použít **padx/pady**, **ipadx**/**ipady** (funkčnost je stejná).

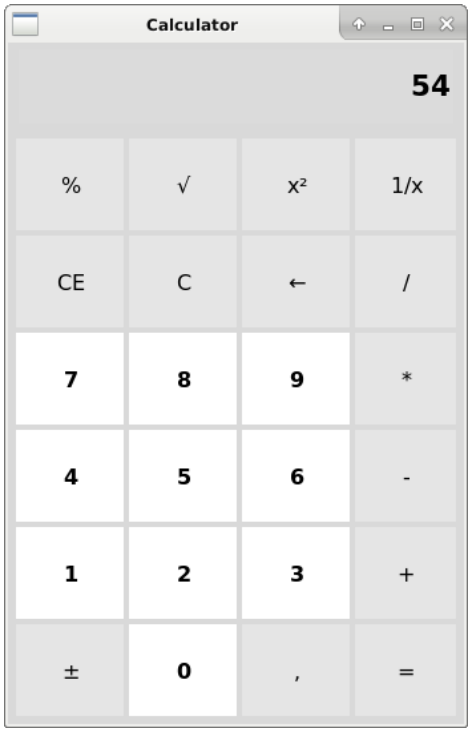
Pozor na kombinaci příkazu **pack** a **grid**.
V rámci jednoho kontejneru není možné kombinovat více manažerů rozvržení. Pokud byste tedy vložili do jednoho kontejneru nějakou komponentu pomoc příkazu **grid** a další pomocí příkazu **pack**, tak se stane to, že se aplikace spustí, ale nic nezobrazí. Pokud se něco takového stane, nemá smysl aplikaci znovu spouštět a je třeba tento problém vyřešit. V případě OS Windows pak i zrušit zbytečně běžící procesy.

Uvnitř jednoho kontejneru tedy nelze pack i grid kombinovat, takže pokud uvnitř kontejneru zvolíte pro vykreslování jeden typ, tak ho musíte použít pro všechny komponenty uvnitř vykreslené. Pokud ale vložíte a vykreslíte uvnitř další kontejner (typicky Frame nebo LabelFrame) můžete uvnitř tohoto kontejneru zvolit libovolný manažer rozvržení, musíte jej samozřejmě použít pro všechny vložené komponenty.

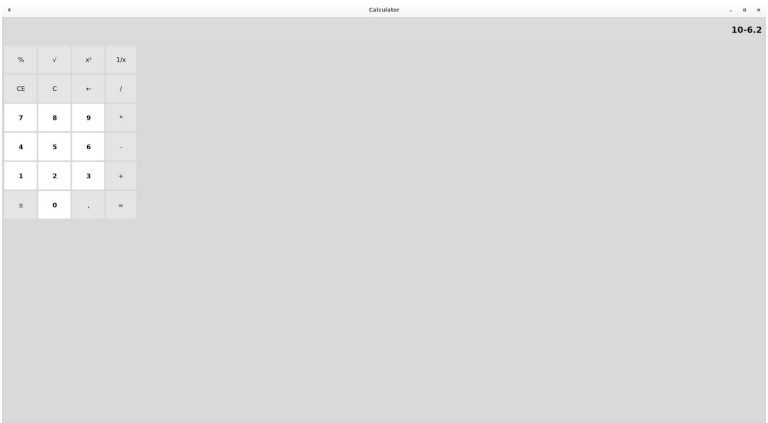
Úkol na cvičení

Jak bylo zmíněno výše, grid rozděluje objekty do mřížky a typickou aplikací, kde takovou mřížku využít je kalkulačka a rozložení tlačítek. Zkuste tedy navrhnout vlastní kalkulačku i se základní funkčností. Rozhraní by mělo dodržovat typické vzhledy kalkulačky (tzn. pozice tlačítek s číslicemi, snažit se mít u sebe tlačítka s podobnými funkcemi), ale jinak můžete zapojit svou fantazii.





V tomto případě již nezamykejte velikost okna a pokuste se vhodně naformátovat jednotlivé komponenty tak, aby při změně velikosti došlo k vhodnému zarovnání a aby nedošlo k něčemu takovému:



Volání funkcí na tlačítkách

Už jste si minule vyzkoušeli volání funkce po stisknutí tlačítka pomocí parametru **command**. U tohoto parametru se předal název funkce, ale bez (), protože by tak došlo rovnou k zavolání funkce. V případě kalkulačky bychom ale chtěli do funkce předat i parametr - nechceme vytvářet funkce onClick1(), onClick2(), atd., ale jen onClick(val). Jak na to? Musíme použít **lambda** funkci.

```
def onClick(param):
        print(param)
b = Button(root, text="URO",command=lambda: onClick("hello"))
```

Nastavení fontu

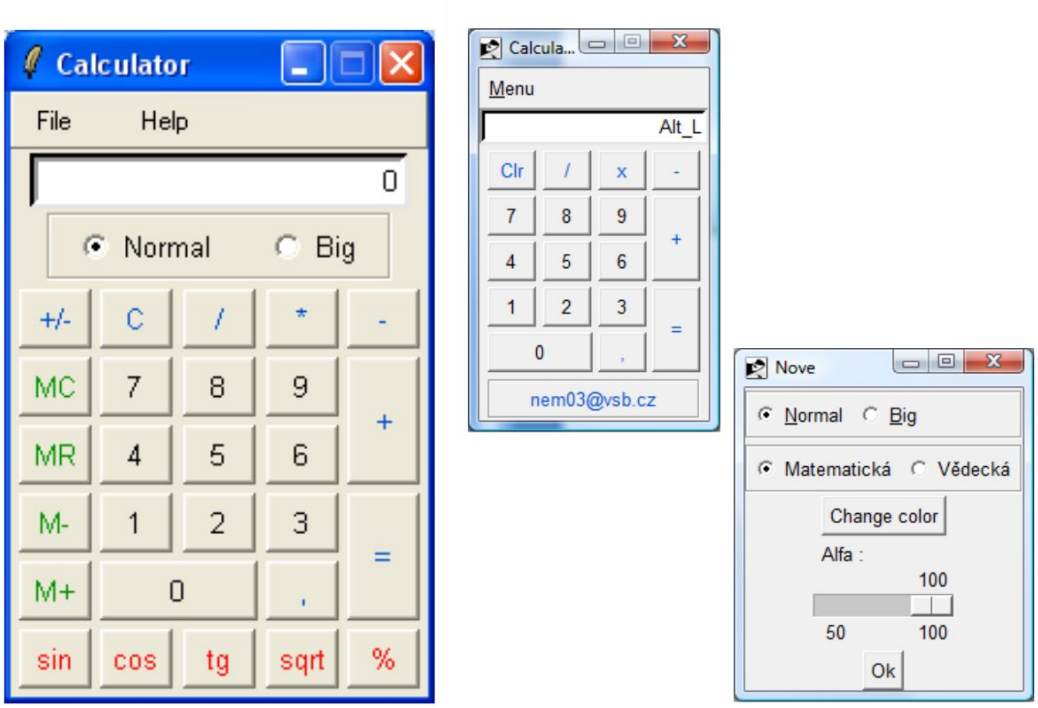
```
from tkinter import font

self.font = font.Font(size=10, weight="normal")
label = Label(root, text="URO", bg="#AAAAAA", font=self.font)
self.font.config(weight="bold", size=16)
```

Výpočet

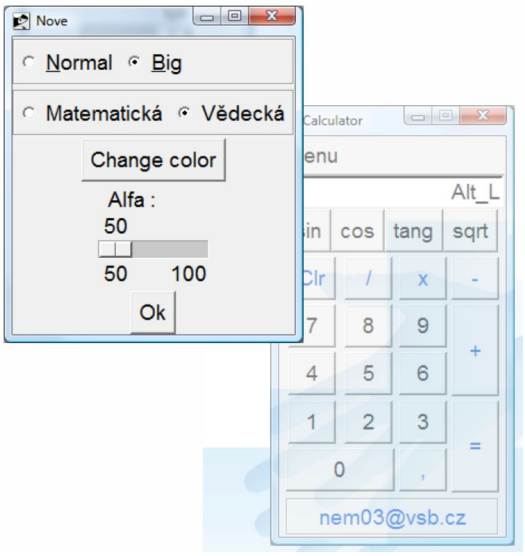
Pro výpočet můžete využít funkci eval, která vyhodnotí zadaný řetězec jako výraz

```
a = "50"
b = "10"
op = "+"
c = eval(a+op+b)
print(c)
```

Výsledný vzhled si už libovolně rozšiřte, přidejte tam menu, vyzkoušejte nové okno s nastavením, přidejte někam své jméno nebo login apod.



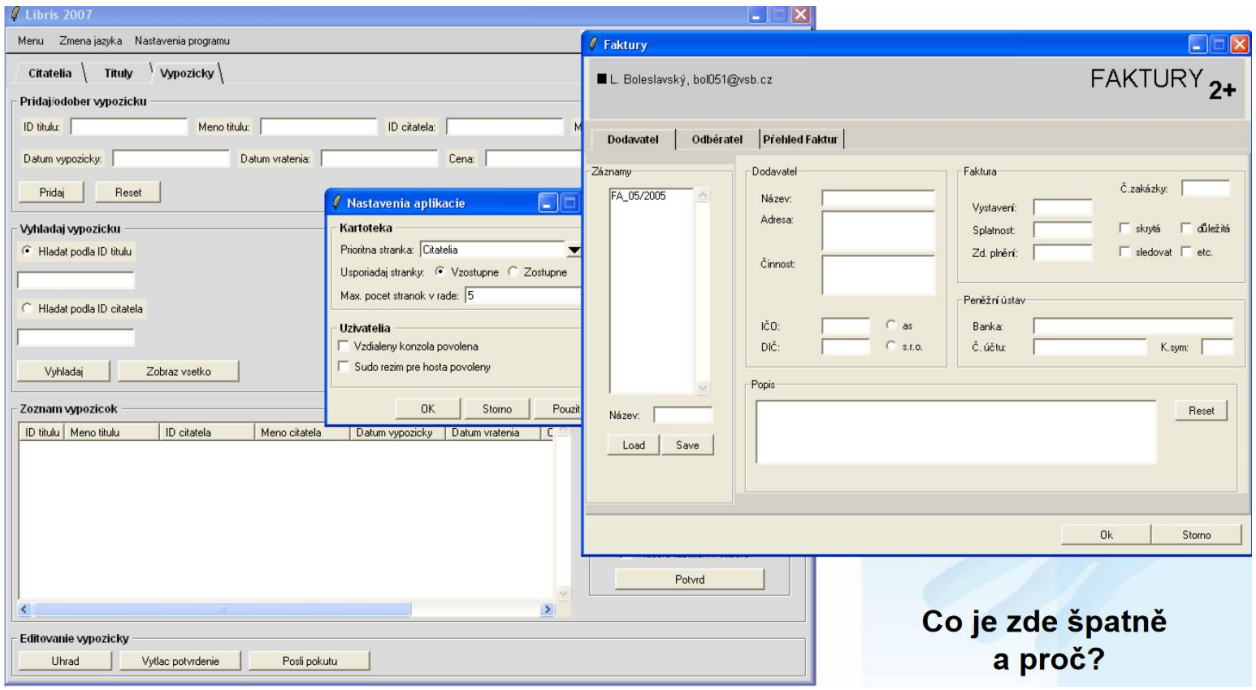
Můžete například přidat v nastavení průhlednost aplikace



root.wm_attributes('-alpha', 0.7)
Objekty se dají nejen dodatečně vytvářet, ale můžete je uvnitř kontejneru i mazat
self.bud.destroy() #zrušení objektu

Návrhy projektů

Nezapomeňte na návrh projektů, kde byste si měli návrh projít z pozice uživatele a přemýšlet, co je v něm správně a co je špatně. Zde patří třeba jak uživatel danou část najde, jak přidá záznam apod. Aplikace by měla mít nějaké komplexnější GUI, aby bylo nutné při realizaci přemýšlet co musí být zobrazeno hned, co lze třeba skrýt do záložek apod.


Libris 2007
Menu
Zmena jazyka
Nastavenia programu
Citatelia
Tituly
Vypozicky
Pridaj/odober vypozicku
ID titulu:
Meno titulu:
ID citatela:
Datum vypozicky:
Datum vratenia:
Cena:
Pridaj
Reset
Vyhladaj vypozicku
Hladat podla ID titulu
Hladat podla ID citatela
Vyhladaj
Zobraz vsetko
Zoznam vypozicok
ID titulu
Meno titulu
ID citatela
Meno citatela
Datum vypozicky
Datum vratenia
Editovanie vypozicky
Uhrad
Vytlac potvrdenie
Posli pokutu
Nastavenia aplikacie
Kartoteka
Prioritna stranka:
Citatelia
Usporiadaj stranky:
Vzostupne
Zostupne
Max. pocet stranok v rade:
5
Uzivatelia
Vzdialeny konzola povolena
Sudo rezim pre hosta povoleny
OK
Storno
Faktury
L. Boleslavský, bol051@vsb.cz
FAKTURY 2+
Dodavatel
Odběratel
Přehled Faktur
Záznamy
FA_05/2005
Název:
Load
Save
Dodavatel
Název:
Adresa:
Činnost:
IČO:
DIČ:
as
s.r.o.
Faktura
Vystavení:
Splatnost:
Zd. plnění:
Č.zakázky:
skrytá
důležitá
sledovat
etc.
Peněžní ústav
Banka:
Č. účtu:
K.sym:
Popis
Reset
Ok
Storno
Potvrd
Co je zde špatně
a proč?