

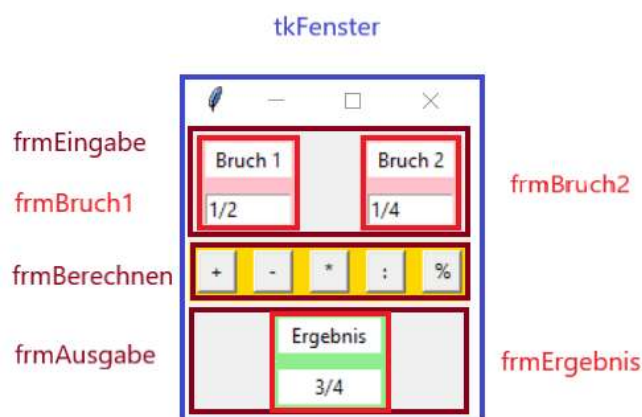
19. GUI-Entwicklung mit tkinter

19.1. Definitionen

- tkinter ist ein GUI-Toolkit zum Erstellen von grafischen Benutzeroberflächen
- eine grafische Benutzeroberfläche (Fenster) besteht aus Steuerelementen
- Steuerelemente sind Bedieneinheiten (z. B. Schaltfläche oder Texteingabefeld)
- Steuerelemente werden auch Widgets oder Controls genannt
- Steuerelemente werden zu einer Benutzeroberfläche zusammengesetzt und mit dem Packer einem Dialog zugeordnet
- jedes Steuerelement hat ein übergeordnetes Element (Master)
- ein Steuerelement darf beliebig viele weitere Steuerelemente enthalten (Slave)

19.2. Steuerelement-Typen

- Beschriftung (Label)
 - zum Anzeigen eines Textes auf der grafischen Benutzeroberfläche
- Schaltfläche (Button)
 - durch Anklicken wird die entsprechende Aktion (command) ausgeführt
- Texteingabefeld (Entry)
 - einzeliliges Eingabefeld
- Rahmen (Frame)
 - zur Strukturierung der grafischen Benutzeroberfläche
 - sie bilden einen „Container“ zur Aufnahme der Steuerelemente
- Beispiel:
 - ein Fenster ist eingeteilt in vier Frames
 - die Frames werden im Fenster angeordnet (z.B. mit dem Layout-Manager pack)
 - in den Frames werden die Steuerelemente angeordnet (z.B. mit dem Layout-Manager pack)
 - zur Strukturierung der grafischen Benutzeroberfläche



19.3. Steuerelemente

19.3.1. Label

- Beschriftung (Label)
 - zum Anzeigen eines Textes auf der grafischen Benutzeroberfläche

Parameter	Bedeutung
master	Zuordnung zum übergeordneten Steuerelement
text	Beschriftung
bg (bzw. background)	Hintergrundfarbe
fg (bzw. foreground)	Textfarbe
Font	Schriftformat

19.3.2. Schaltfläche (Button)

- Schaltfläche (Button)
 - durch Anklicken wird die entsprechende Aktion (command) ausgeführt

Parameter	Bedeutung
Master	Zuordnung zum übergeordneten Steuerelement
Text	Beschriftung
bg (bzw. background)	Hintergrundfarbe
fg (bzw. foreground)	Textfarbe
font	Schriftformat
command	Anbinden der Ereignisbehandlungsprozedur wird beim Anklicken ausgeführt

19.3.3. Texteingabefeld (Entry)

- Texteingabefeld (Entry)
 - einzeliges Eingabefeld

Parameter	Bedeutung
master	Zuordnung zum übergeordneten Steuerelement
bg (bzw. background)	Hintergrundfarbe
fg (bzw. foreground)	Textfarbe
font	Schriftformat

19.3.4. Rahmen (Frame)

- Rahmen (Frame)
 - zur Strukturierung der grafischen Benutzeroberfläche
 - sie bilden einen „Container“ zur Aufnahme der Steuerelemente

-

Parameter	Bedeutung
master	Zuordnung zum übergeordneten Steuerelement
bg (bzw. background)	Hintergrundfarbe
fg (bzw. foreground)	Textfarbe
font	Schriftformat

-

19.4. Layout-Manager

- der place-Manager benutzt Koordinatensysteme, um GUI-Komponenten zu platzieren
- der pack-Manager packt alle GUI-Komponenten in ein System aus ineinander geschachtelten Rahmen
- der grid-Manager benutzt ein Raster, in das alle GUI-Komponenten eingepasst werden.

19.4.1. Layout-Manager pack()

- Der pack-Manager (bzw. Packer) bestimmt die Anordnung von GUI-Komponenten automatisiert unter Berücksichtigung von Vorgaben im Programm

Parameter	Bedeutung
side	Seite der Master-Komponente, an die die GUI-Komponente gesetzt wird mögliche Werte: 'left', 'right', 'top', 'bottom'
padx, pady	Leerer Platz rechts und links bzw. ober- und unterhalb der GUI-Komponente
fill	die GUI-Komponente wird so vergrößert, dass sie den gesamten zur Verfügung stehenden Platz ausfüllt mögliche Werte: 'x', 'y', 'both'.

19.4.1.1. Benutzeroberfläche mit Layout-Manager pack()

- Beispiel einer Benutzeroberfläche für die Klasse Bruch

```
from tkinter import *

# Einbinden der Klasse Bruch
from clBruch import Bruch

# Instanziiieren der Objekte der Klasse Bruch
bruch1 = Bruch()
bruch2 = Bruch()

def btnPlusClick():
    # Übernahme der Daten
    # Verarbeitung der Daten
    # Anzeige der Daten

def btnMinusClick():
    # Übernahme der Daten
    # Verarbeitung der Daten
    # Anzeige der Daten

def btnMultClick():
    # Übernahme der Daten
    # Verarbeitung der Daten
    # Anzeige der Daten

def btnDivClick():
    # Übernahme der Daten
    # Verarbeitung der Daten
    # Anzeige der Daten

def btnRestClick():
    # Übernahme der Daten
    # Verarbeitung der Daten
    # Anzeige der Daten

# Fenster/Window
tkFenster = Tk()
tkFenster.title('Rechner')

# Frames
frmEingaben = Frame(master=tkFenster)
frmBruch1 = Frame(master=frmEingaben, bg='pink')
frmBruch2 = Frame(master=frmEingaben, bg='pink')

frmBerechnen = Frame(master=tkFenster, bg='gold')

frmAusgaben = Frame(master=tkFenster)
frmErgebnis = Frame(master=frmAusgaben, bg='lightgreen')

frmEingaben.pack(fill='x')
frmBerechnen.pack()
frmAusgaben.pack(fill='x')
frmErgebnis.pack(padx='5', pady='5')
frmBruch1.pack(side='left', padx='5', pady='5')
frmBruch2.pack(side='right', padx='5', pady='5')
```

```

# lbl mit Text Bruch 1
lblBruch1 = Label(master=frmBruch1, bg='white', text='Bruch 1')
lblBruch1.pack(padx='5', pady='5', fill='x')
# Entry für Bruch 1
entBruch1 = Entry(master=frmBruch1, bg='white', width='8')
entBruch1.pack(padx='5', pady='5')

# lbl mit Text Bruch 2
lblbruch2 = Label(master=frmBruch2, bg='white', text='Bruch 2')
lblbruch2.pack(padx='5', pady='5', fill='x')
# Entry für Bruch 2
entbruch2 = Entry(master=frmBruch2, bg='white', width='8')
entbruch2.pack(padx='5', pady='5')

# btn zum Addieren
btnPlus = Button(master=frmBerechnen, text='+', width='2',
command=btnPlusClick)
btnPlus.pack(side='left', padx='5', pady='5')

# btn zum Subtrahieren
btnMinus = Button(master=frmBerechnen, text='-',
width='2',command=btnMinusClick)
btnMinus.pack(side='left', padx='5', pady='5')

# btn zum Multiplizieren
btnMult = Button(master=frmBerechnen, text='*', width='2',
command=btnMultClick)
btnMult.pack(side='left', padx='5', pady='5')

# btn zum Dividieren ohne Rest
btnDiv = Button(master=frmBerechnen, text=':', width='2',
command=btnDivClick)
btnDiv.pack(side='left', padx='5', pady='5')

# btn zum Rest bei der Division
btnRest = Button(master=frmBerechnen, text='%', width='2',
command=btnRestClick)
btnRest.pack(side='left', padx='5', pady='5')

# lbl mit Text Ergebnis
lblTextErgebnis = Label(master=frmErgebnis, bg='white',
text='Ergebnis')
lblTextErgebnis.pack(padx='5', pady='5', fill='x')

# lbl für das Ergebnis
lblErgebnis = Label(master=frmErgebnis, bg='white', width='8',
text='')
lblErgebnis.pack(padx='5', pady='5', fill='x')

# Aktivierung des Fensters
tkFenster.mainloop()

```