

GUI

Pygame

- gut für Spiele mit vielen Grafiken und Musik/Sounds.
- Elemente Animieren.
- Einbinden von Tastatur, Maus und Gamepads.
- Kollisionskontrolle

tkinter

- Das Standard Python Interface.
- am häufigsten benutzt.
- Auf den meisten Plattformen verfügbar.

guizero

- geeignet für Anfänger, schnell zu lernen.
- benutzt die tkinter GUI library(man muss keine anderen installieren).
- Flexibel genug um Qualitative Programme zu entwickeln.
- Gute Dokumentation mit Beispielen vorhanden.
- zeigt weitere Hilfreiche Fehlermeldungen an.
- simplere Syntax als tkinter.

Fazit

PyGame ist gut für Spiele und nicht wirklich für Benutzeroberflächen, es ist gut dafür Grafiken und Sounds zu verarbeiten sowie Animationen und Inputs.

Tkinter ist das am meisten verwendete GUI und schnell zu lernen. Es ist gut für Benutzeroberflächen und Anwendungen mit simpler Grafik.

Guizero ist ein Wrapper für Tkinter und etwas simpler gestaltet um es schnell zu verstehen mit einer guten Dokumentation mit vielen Beispielen.

Faktoren

PyGame

- + Gutes Spieldesign(Grafiken, Sounds, Animationen)
- + Tastatur, Maus und Gamepads einbinden
- Nicht optimal für Simple Benutzeroberflächen
- Nicht mit Tkinter/guizero Kompatibel
- Es gibt keine Buttons oder ähnliche Eingabe Elemente
- Ohne Librarys keine Layout-control außer durch width/height

Tkinter

- + Schnell zu lernen
- + Optimal für simple Benutzeroberflächen
- + Auf den meisten Plattformen verfügbar
- Nicht optimal für Spiele
- = Pack, Grid und Place Layout-control

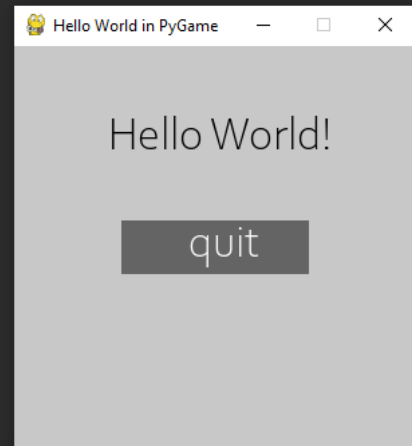
guizero

- + Wie Tkinter nur Simpler(Wrapper für Tkinter)
- + Es muss nichts weitere Installiert werden
- + Gute Dokumentation
- + zeigt Hilfreiche Fehlermeldungen an
- Nicht optimal für Spiele
- Elemente unpräzise platzierbar
- = Auto, Align, Filling und Grid Layout-control

Syntax Vergleich

PyGame

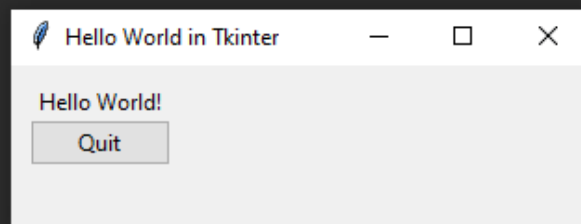
```
1 import pygame
2
3 pygame.init()
4 pygame.display.set_caption('Hello World in PyGame')
5 # screen resolution
6 res = (300,300)
7 # opens up a window
8 screen = pygame.display.set_mode(res)
9
10 width = screen.get_width()
11 height = screen.get_height()
12 # defining a font
13 smallfont = pygame.font.SysFont('Corbel',35)
14 # rendering a text written in this font
15 text = smallfont.render('quit' , True , (255,255,255))
16 world = smallfont.render('Hello World!' , True , (0,0,0))
17
18 while True:
19     for event in pygame.event.get():
20         if event.type == pygame.QUIT:
21             pygame.quit()
22         #checks if a mouse is clicked
23         if event.type == pygame.MOUSEBUTTONDOWN:
24             #if the mouse is clicked on the
25             # button the game is terminated
26             if width/2-70 <= mouse[0] <= width/2 + 70 and height/2-20 <= mouse[1] <= height/2+20:
27                 pygame.quit()
28
29     # fills the screen with a color
30     screen.fill((200,200,200))
31     # stores the (x,y) coordinates into
32     # the variable as a tuple
33     mouse = pygame.mouse.get_pos()
34     screen.blit(world , (width/2-80,height/2-100))
35     pygame.draw.rect(screen,(100,100,100),[width/2-70,height/2-20,140,40])
36     # superimposing the text onto our button
37     screen.blit(text , (width/2-20,height/2-20))
38     # updates the frames of the game
39     pygame.display.update()
```



Tkinter

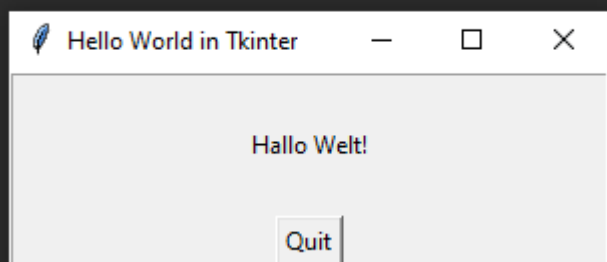
Grid control

```
1 from tkinter import *
2 from tkinter import ttk
3 root = Tk()
4 frm = ttk.Frame(root, padding = 10)
5
6 def action():
7     root.destroy()
8
9 frm.grid()
10
11 root.title("Hello World in Tkinter")
12
13 ttk.Label(frm, text = "Hello World!").grid(column=0, row=0)
14
15 ttk.Button(frm, command = action, text ="Quit").grid(column=0, row=1)
16
17 root.mainloop()
```



Pack control

```
1 import tkinter
2
3 app = tkinter.Tk()
4 app.geometry("300x100")
5 app.title("Hello World in Tkinter")
6 frame = tkinter.Frame(app, relief="ridge", borderwidth=2)
7 frame.pack(fill="both", expand=1)
8 label = tkinter.Label(frame, text="Hallo Welt!")
9 label.pack(expand = 1)
10 button = tkinter.Button(frame, text="Quit", command=app.destroy)
11 button.pack(side="bottom")
12
13 app.mainloop()
```



GUI Zero

```
1 from guizero import App, Text, PushButton
2
3 app = App("Hello World in GUI Zero")
4
5 def action():
6     app.destroy()
7 text = Text(app, "Hello World!")
8
9 button = PushButton(app, action, text="Quit")
10
11 app.display()
```

