



Steuerelementvariablen (Control variables) und Events in Tkinter

Lernziele:

- Verständnis der Steuerelementvariablen in Tkinter (`StringVar`, `IntVar`, `DoubleVar`, `BooleanVar`)
- Nutzung von `trace()`, um Änderungen an Variablen zu erkennen
- Einführung in das **Eventhandling** mit `command` und `bind()`



Steuerelementvariablen (Control variables) und Events in Tkinter

Steuerelementvariablen (Control Variables):

In der Tkinter-Bibliothek von Python sind **Steuerelementvariablen (Control Variables)** spezielle Objekte wie **StringVar**, **IntVar**, **DoubleVar** oder **BooleanVar**, die mit Widgets wie Entry-Feldern, Kontrollkästchen oder Optionsfeldern verbunden werden können.

Sie ermöglichen es, den aktuellen Wert eines Widgets zu verfolgen und das Widget automatisch zu aktualisieren, sobald sich der Wert ändert – entweder durch den Benutzer oder programmgesteuert.

Diese Variablen sind Instanzen der Tkinter-Klassen StringVar, IntVar, DoubleVar und BooleanVar, die für verschiedene Datentypen entwickelt wurden.



Steuerelementvariablen (Control variables) und Events in Tkinter

Hier ist eine kurze Übersicht über jeden Typ von Kontrollvariablen und wie man sie verwendet:

- **StringVar**: Wird für Text- oder Zeichenketten-Daten (`string`) verwendet.
- **IntVar**: Wird für ganzzahlige Daten (`integer`) verwendet.
- **DoubleVar**: Wird für Gleitkomma-Daten (`float`) verwendet.
- **BooleanVar**: Wird für boolesche Daten (`True/False`) verwendet.



Steuerelementvariablen (Control variables) und Events in Tkinter

Code Beispiel

```
import tkinter as tk

# Funktion zur Anzeige des aktuellen Werts
# .get() liest den aktuellen Zustand der BooleanVar:
def zeige_status():
    print("Checkbox ausgewählt:", zustand.get())

root = tk.Tk()
root.title("BooleanVar Beispiel mit Checkbutton")

# Was ist eine BooleanVar?
# Eine BooleanVar ist eine spezielle Variable in Tkinter,
# die nur zwei Werte speichern kann: True oder False.
# Sie wird z.B. bei Checkboxes verwendet, um zu merken, ob diese angeklickt wurden oder nicht.
zustand = tk.BooleanVar()
zustand.set(False) # Anfangswert: nicht ausgewählt

# Checkbutton erstellt, verbunden mit der BooleanVar
check = tk.Checkbutton(root, text="Ich bin einverstanden", variable=zustand, command=zeige_status)

check.pack(padx=20, pady=10)

root.mainloop()
```



Steuerelementvariablen (Control variables) und Events in Tkinter

Nachverfolgen von Variablen

Sie können Änderungen an diesen Variablen mit der **trace**-Methode verfolgen, die eine Rückruffunktion aufruft, wann immer die Variable gelesen, beschrieben oder gelöscht wird.

```
import tkinter as tk
```

```
def on_change(*args):  
    print("Variable geändert:", entry_var.get())  
    # Es gibt die Benutzereingaben im Textfeld aus
```

```
root = tk.Tk()
```

```
entry_var = tk.StringVar()  
entry_var.trace("w", on_change)
```

```
entry = tk.Entry(root, textvariable=entry_var)  
entry.pack()
```

```
root.mainloop()
```



Steuerelementvariablen (Control variables) und Events in Tkinter

Eventhandling in Tkinter

In grafischen Benutzeroberflächen (GUIs) lösen Benutzeraktionen wie Mausklicks, Tastendrucke oder das Bewegen der Maus sogenannte **Events (Ereignisse)** aus.

Damit ein Programm sinnvoll auf solche Aktionen reagieren kann, müssen diese Ereignisse erkannt und verarbeitet werden – dieser Vorgang wird **Eventhandling** genannt.

Tkinter stellt dafür **zwei Hauptmethoden** bereit: **command** für einfache Aktionen bei Buttons oder Menüeinträgen, und **bind()** für detaillierte Ereignissteuerung wie Tastatureingaben oder Mausklicks.

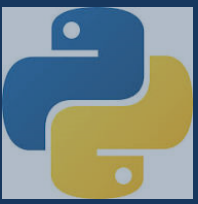
Methode	Beschreibung	Beispiel
<code>command</code>	Wird direkt im Widget (z. B. Button) gesetzt	<code>Button(..., command=func)</code>
<code>bind()</code>	Reagiert auf bestimmte Events (Tasten, Maus etc.)	<code>widget.bind("<Key>", func)</code>



Steuerelementvariablen (Control variables) und Events in Tkinter

Eventhandling in Tkinter

```
def on_click():  
    print("Button wurde geklickt!")  
  
button = tk.Button(root, text="Klick mich", command=on_click)  
button.pack()  
  
def on_key(event):  
    print("Taste gedrückt:", event.char)  
  
root.bind("<Key>", on_key)  
# event.char enthält das Zeichen der gedrückten Taste, z.B. "a" oder "B"  
# Für Sondertasten wie Shift oder Pfeiltasten ist event.char leer  
# beispieldruck: Taste gedrückt: B
```



Steuerelementvariablen (Control variables) und Events in Tkinter

Mehr erkunden:

<https://www.geeksforgeeks.org/python-setting-and-retrieving-values-of-tkinter-variable/>

<https://anzelg.github.io/rin2/book2/2405/docs/tkinter/control-variables.html#:~:text=A%20Tkinter%20control%20variable%20is,as%20a%20number%20or%20string.>

<https://www.geeksforgeeks.org/python/python-binding-function-in-tkinter/>

<https://dafarry.github.io/tkinterbook/tkinter-events-and-bindings.htm>

<https://www.codersarts.com/forum/python-tkinter/tkinter-tutorial-4-events-and-bindings>

<https://anzelg.github.io/rin2/book2/2405/docs/tkinter/binding-levels.html>

<https://www.pythontutorial.net/tkinter/tkinter-event-binding/>