Informatik Klasse 13, Foliensatz 6 Wiederholung, erste Beispiele für Ereignisse und Bindungen

Prof. G. Kemnitz

Institut für Informatik, Technische Universität Clausthal

1. Dezember 2010

Wiederholung Widgets

Wozu dienen die folgenden Widgets? Wie sehen sie aus?

- Button
- Label
- Message
- Frame
- Entry

Welche Methoden hat jede der fünf Widget-Klassen?
Für welche Widget-Klasse sind die anderen Methoden nur definiert?
Welche Attribute haben alle diese Widget-Klassen?
Für welche Widget-Klasse sind die übrigen Attribute nur definiert?

Wiederholung Methoden

```
w.config(...)
w.get(...)
w.cget(...)
w.pack(...)
w.grid(...)
```

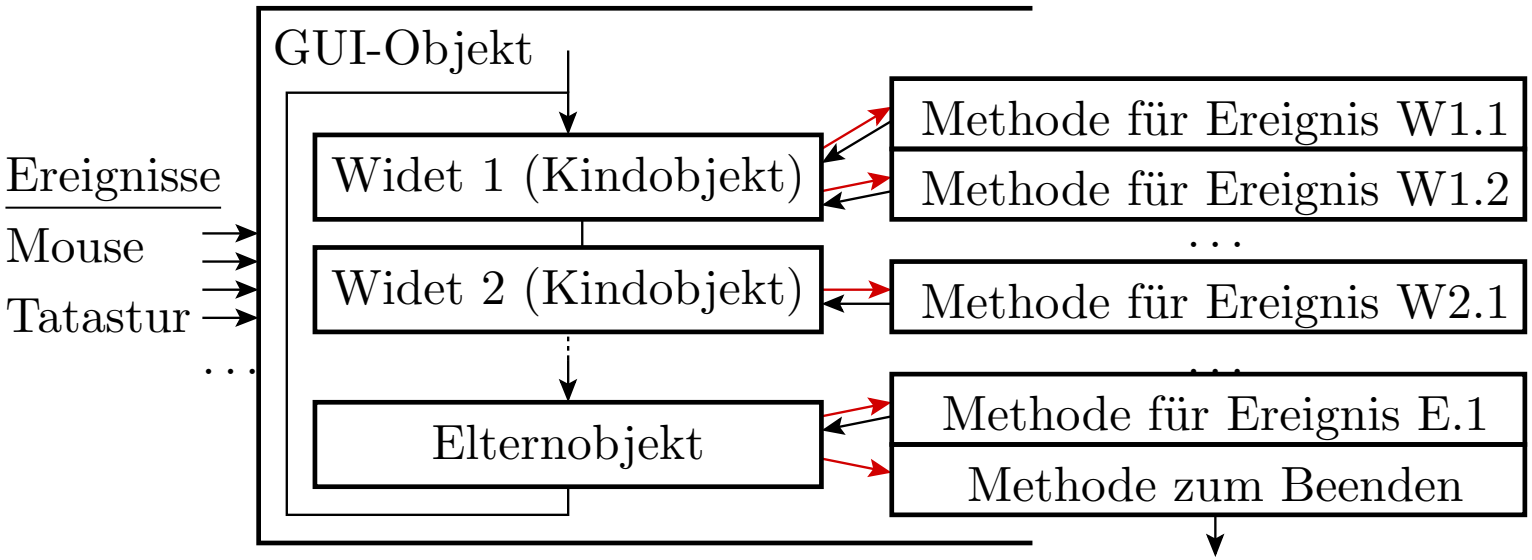
Was bewirken die Methoden?
Welche Widget.Klassen sind für »w« für die einzelnen Methoden erlaubt?
Welche Aufrufparameter sind erforderlich bzw. zulässig?
Welchen Typ hat der Funktionswert?

Wiederholung Attribute

```
bg
font
text
cursor
xpos
command
```

Was bewirken die Attribute?
Für welche Widget-Klassen sind sie definiert?
Welchen Datentyp haben die Attribute?
Wie können die Attribute verändert werden?

Die Methode »bind«



- Zuordnung von Funktionsaufrufen zu Ereignissen

  widget`.bind(`*Ereignis*`, `*Ereignishandler*`)`

Widget – Widget-Objekt
Ereignis – Mouse-Bewegung, Tastatureingabe etc.
Ereignishandler – aufzurufende Funktion/Methode

Experiment mit Mouse-Ereignissen

```
from Tkinter import *
def Ereignishandler(EreignisObj):
 print "clicked at", EreignisObj.x, EreignisObj.y
root = Tk()
frame = Frame(root, width=100, height=100)
frame.bind("<Button-1>", Ereignishandler)
frame.pack()
root.mainloop()
```

- width, height – Attribute zur Festlegung der Fenstergröße
- "<Button-1>" – Bezeichner für das Ereignis
- EreignisObj.x, EreignisObj.y – Attribute des Ereignisobjekts, hier die relative Mouse-Position

Aktionsbindung an die Eingabetaste

```
from Tkinter import *

def Ereignishandler(EreignisObj):
 print "Eingabezeichen:", EreignisObj.widget.get()

root = Tk()
E = Entry(root)
E.bind("<Return>", Ereignishandler)
E.pack()
root.mainloop()
```

- "<Return>" Ereignis »Return-Taste gedrückt«
- *EreignisObj*.widget – Referenz auf Widget, das das Ereignis verursacht hat
- *EingabeWidget*.get() Methode zum Lesen des Eingabewertes

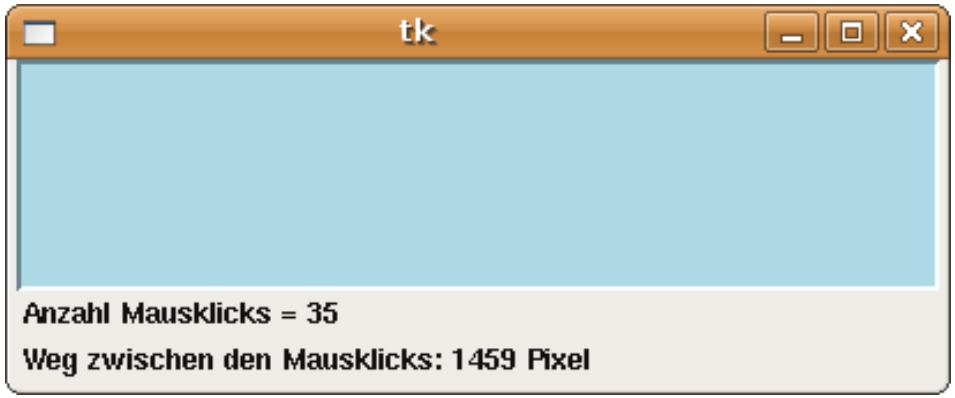
Experiment »Label als Button«

```
from Tkinter import *
def Ereignishandler(EreignisObj):
 print "clicked at", EreignisObj.widget.cget('text')
root = Tk()
for idx in range(5):
 x=Label(root, text=str(idx), width=20,
   borderwidth=2, relief="sunken")
 x.bind("<Button-1>", Ereignishandler)
 x.pack()
root.mainloop()
```

- relief – Attribut für die Umrandung
- borderwidth – Attribute für die Dicke der Umrandung

Aufgabe 6.1: Wegeberechnung

Schreiben Sie eine Applikation mit folgender Oberfläche, das die Anzahl der Mausklick in dem farbig unterlegten Frame zählt und den zurückgelegten Weg zwischen den Mouse-Klicks aufsummiert.

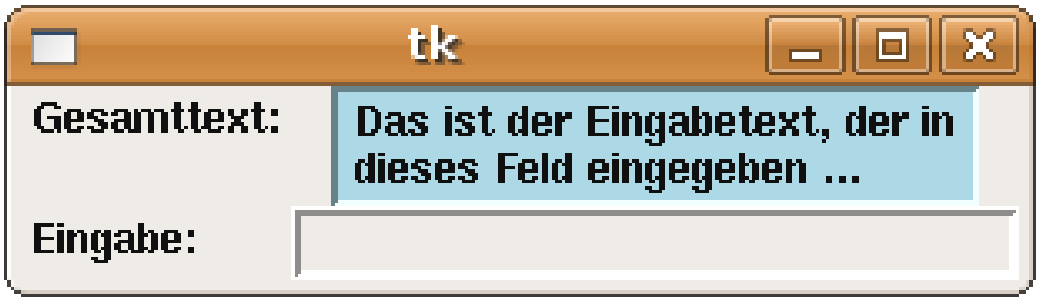


Hinweise:

- Ereignisbindung wie im Experiment »Mouse-Ereignis«
- Größe des farbig unterlegten Frames 400 mal 100 Pixel.
- Die beiden Ausgabezeilen unten seinen Labels. Ausgabe als Zeichenkette berechnen und mit »config« zuweisen
- Linksausrichten der Ausgabe mit »*l*.pack(anchor='w')«

Aufgabe 6.2: Verbesserte GUI-Erweiterung der Text-Klasse

Schreiben Sie eine Textverarbeitungsanwendung mit folgender Oberfläche, bei der eingegebene Text nach Betätigung der Eingabetaste an den Text des Message-Objekts oben angehängt wird.



- Ereignisbindung wie im Experiment »Aktionsbindung an die Eingabetaste«
- Anordnung mit »grid«; Ausrichtung innerhalb eines Feldes mit »sticky='nw'«, z.B.
- Label(root, text='Text').grid(row=0, sticky='nw')

Aufgabe 6.3: Tastenfeld zur Zahleneingabe

Schreiben Sie eine Anwendung zur Zahleneingabe, bei der die angeklickte Zahl hinten an den Eingabewert angehängt wird mit folgender Oberfläche:

- Ereignishandler wie im Experiment »Label als Button«

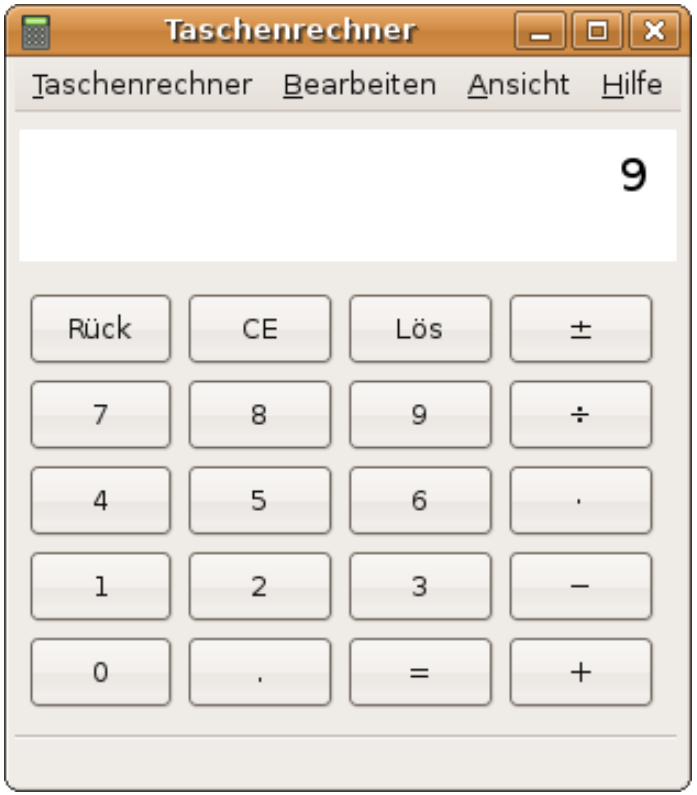
- Erzeugung der Label-Matrix in einer verschachtelten Schleife; Label-Beschriftung als D-Tupel:

```
t=(('0', '1', '2', '3'), ('4', '5', '6', '7'),
   ('8', '9', 'A', 'B'), ('C', 'D', 'E', 'F'))
```

- Tastenfeld in ein Frame einrahmen; Frame und Ausgabe-Label mit »pack« übereinanderstapeln

Aufgabe 6.4: Taschenrechner

Schreiben Sie nach dem Schema in der Aufgabe zu vor einen Taschenrechner mit den Grundrechenarten und einer Oberfläche ähnlich der Nachfolgenden, nur ohne Pull-Down-Menüs.



Aufgabe 6.5: Puzzel 1

Es gab früher eine Art Puzzel-Spiel aus 24 mechanisch verschiebbaren Teilen und einem Leerfeld, bei dem man durch Verschieben Zahlen oder Buchstaben sortieren konnte.
Programmieren Sie für dieses Spiel einer 5×5-Label-Matrix mit den Ziffern 1 bis 24 und einem Leerfeld:

Bei einem Maus-Klick auf ein Label neben dem Leerfeld soll das Label in das Leerfeld springen. Sie benötigen hierfür folgende bisher noch nicht eingeführte Methoden des Geometrie-Managers »grid«:

```
widget.grid_info()
```

liefert ein Wörterbuch aller Attribute des Geometrie-Managers, u.a. die Zeilen- und die Spaltenummer.

```
widget.grid_configure(row=..., column=...)
```

erlaubt die Attribute zu ändern.

Aufgabe 6.6: Puzzel 2

Erweitern Sie das Programm aus der Aufgabe zuvor

- um eine Taste, bei deren Betätigung die Zifferen-Label und das Leerfeld an zufälligen Positionen angeordnet werden, und
- so, dass beim Anklicken eines entfernten Labels in einer Reihe oder Spalte mit dem Leerfeld die Label-Reihe bzw. Spalte ab dem angeklickten Label in Richtung Leerfeld verschoben wird.

Hinweis:

- Ein Label, dass auf ein besetztes Feld verschoben wird, bleibt unsichtbar.

Aufgabe 6.7: Puzzel 3

Erweitern Sie das Programm aus der Aufgabe zuvor

- um eine Taste »random«, bei deren Betätigung ein zufälliger Spielzug, der eine Verschiebung bewirkt, ausgeführt wird
- eine »undo«-Liste mit allen Spielzügen
- eine Taste »undo«, bei deren Betätigung der letzte getätigte Spielzug rückgängig gemacht wird, und
- ein Label, dass die Anzahl der getätigten/rückgängig machbaren Spielzüge anzeigt.

Aufgabe 6.8: Vokabeltrainer 1

Entwickeln Sie einen Vokabeltrainer mit

- einem Wörterbuch, das mindestens 10 deutschen Wörtern als Schlüssel eine Liste möglicher englischer Übersetzungen zuordnet.
- nach Betätigung der Taste Start fünf mal nacheinander
  - in einem Label einen zufällig ausgewählten Schlüssel des Wörterbuchs ausgibt
  - zur Eingabe eines zugeordneten englischen Wortes auffordert
  - nach Betätigung von <Enter> die Eingabe mit den möglichen Übersetzungen vergleicht
  - bei einer richtigen Eingabe einen Punktezähler in einem Label erhöht.
- Entwerfen Sie eine geeignete Benutzeroberfläche.

Aufgabe 6.9: Vokabeltrainer 2

Erweitern Sie das Programm aus der vorherigen Aufgabe

- um eine Möglichkeit zur Eingabe neuer Wörterbucheinträge
- eine Möglichkeit zur Bearbeitung vorhandener Einträge
- einer Initialisierung des Wörterbuches aus einer Datei zum Programmstart und einer Abspeicherung in eine Datei zum Programmende.

Aufgabe 6.10: Vokabeltrainer 3

Erweitern Sie das Programm aus der vorherigen Aufgabe um eine Umschaltmöglichkeit »deutsch-englisch« nach »englisch-deutsch«.

Das Wörterbuch mit den englischen Schlüsseln soll aus dem Wörterbuch mit deutschen Schlüsseln erzeugt werden, indem jedem englischen Wort aus den Eintragslisten eine Liste aller deutschen Worte zugeordnet wird, in deren Eintragsliste es im anderen Wörterbuch enthalten ist.