

Tema 20.

• Evenimente în Tkinter

Ce ne așteaptă?

1. **Noțiune de eveniment**
2. **Evenimente asociate mouse-ului și cursorului**
3. **Evenimente asociate tastaturii**

- 1. Ce reprezintă un eveniment?**
- 2. Ce reprezintă funcția de manipulare a evenimentului?**
- 3. Cum se atribuie un eveniment unei componente?**
- 4. Care sunt evenimentele asociate butoanelor mouse-ului?**
- 5. Care sunt evenimentele asociate cursorului?**
- 6. Care sunt evenimentele asociate butoanelor tastaturii?**
- 7. Cum se notează un eveniment asociat tastelor speciale?**
- 8. Cum se notează un eveniment asociat tastelor normale?**
- 9. Cum se notează un eveniment asociat combinației de taste?**

1. Noțiune de eveniment

Evenimente și atribuirea lor

- Evenimentul – acțiune ce pot fi aplicate componentelor și care au drept efect apelarea unei funcții de manipulare a evenimentului
- Exemplu de eveniment: tastarea butonului stâng al mouse-ului, intrarea cursorului pe o componentă, tastarea unui buton de la tastatură, etc
- Metoda `bind()` - atribuie evenimentul unei componente:

`componenta.bind(eventiment, funcție_manipulare)`

- După atribuirea evenimentului către o componentă, atunci când se vor realiza acțiunile respectivului eveniment se va apela o funcție de manipulare a evenimentului

Exemplu de eveniment

- Aplicație care schimbă culoarea ferestrei de baza în galben atunci când cursorul se află pe spațiul ferestrei, în albastru atunci când cursorul părăsește spațiul ferestrei și în roșu atunci când se apasă un buton al mouse-ului pe suprafața ferestrei

```
from tkinter import *
root=Tk()
root.geometry('300x200')

def culoare_rosie(event):
    root.config(bg='red')

def culoare_galbena(event):
    root.config(bg='yellow')

def culoare_albastru(event):
    root.config(bg='blue')

root.bind('<Button>', culoare_rosie)
root.bind('<Enter>', culoare_galbena)
root.bind('<Leave>', culoare_albastru)

root.mainloop()
```

Lista de evenimente asociate mouse-ului

- **<Button>** – se apasă orice buton al mouse-ului pe componentă
- **<Button-1>** – se apasă buton stâng al mouse-ului pe componentă
- **<Button-2>** – se apasă buton din centru al mouse-ului pe componentă
- **<Button-3>** – se apasă buton drept al mouse-ului pe componentă
- **<ButtonRelease>** – se eliberează orice buton al mouse-ului apăsat pe componentă
- **<Motion>** – se mișcă cursorul pe componentă
- **<Enter>** – se introduce cursorul pe spațiul componentei
- **<Leave>** cursorul părăsește spațiul componentei
- **<MouseWheel>** se activează butonul rotativ al mouse-ului pe componentă
- **<Double-Button>** se apasă dublu click cu orice buton al mouse-ului pe componentă

Lista de evenimente asociate tastaturii

- **<Key>** – se apasă orice buton al tastaturii
- **<KeyPress>** – se apasă orice buton al tastaturii (în momentul apăsării)
- **<KeyRelease>** – se eliberează orice buton apăsat al tastaturii (în momentul eliberării)
- **Apăsarea tastelor speciale** – numele tastelor în <>: F1, F2, ..., F12, Left, Right, Up, Down, BackSpace, Delete, End, Home, Insert, Print, Tab
- **<Escape>** – se apară butonul ESC
- **<Return>** – se apară butonul Enter
- **Apăsarea tastelor normale** – numele tastelor fără <>: a, A, +
- **Apăsarea combinațiilor de taste:** <Alt-a>, <Control-Up> etc.

2. Evenimente asociate mouse-ului și cursorului

Evenimentul <Button>

- Apelează funcția de manipulare a evenimentului atunci când se apasă orice buton al mouse-ului pe componenta căreia este atribuit

```
from tkinter import *  
root=Tk()root.geometry('400x400')  
  
def mouseClick(event):  
    text.pack_forget()  
    text.pack(fill = BOTH, expand = True)  
    if text.cget('bg')== "gold":  
        text.config(bg="silver")  
    else:  
        text.config(bg="gold")  
  
text=Label(text="Ai dat click")  
root.bind('<Button>', mouseClick)  
  
root.mainloop()
```


Atributele obiectului evenimentului <Button>

- Evenimentul <Button> creează un obiect al evenimentului care este transmis în mod automat funcției de manipulare a acestuia. Acest obiect are 2 atributele: x și y care specifică coordonatele punctului unde s-a dat click

```
from tkinter import *
root=Tk()
root.geometry('400x400')

def mouseClick(event):
    text.pack_forget()
    text.pack(fill = BOTH, expand = True)
    text.config(text=f"Ai dat click pe punctul ({event.x}, {event.y})")
    if text.cget('bg')=="gold":
        text.config(bg="silver")
    else:
        text.config(bg="gold")

text=Label()
root.bind('<Button>', mouseClick)
root.mainloop()
```

Evenimentele <Button-1>, <Button-2> și <Button-3>

- Apelează funcția de manipulare a evenimentului atunci când se apasă buton stâng (<Button-1>), central (<Button-2>) sau drept (<Button-3>) al mouse-ului pe componenta căreia este atribuit

```
from tkinter import *
root=Tk()
root.geometry('400x400')
def click_stang(event):
    text.pack_forget()
    text.pack(fill = BOTH, expand = True)
    text.config(text=f"Ai dat click stang pe punctul ({event.x}, {event.y})")
    text.config(bg="silver")
def click_drept(event):
    text.pack_forget()
    text.pack(fill = BOTH, expand = True)
    text.config(text=f"Ai dat click drept pe punctul ({event.x}, {event.y})")
    text.config(bg="gold")
def click_central(event):
    text.pack_forget()
    text.pack(fill = BOTH, expand = True)
    text.config(text=f"Ai dat click central pe punctul ({event.x}, {event.y})")
    text.config(bg="purple")
text=Label()
root.bind('<Button-1>', click_stang)
root.bind('<Button-2>', click_central)
root.bind('<Button-3>', click_drept)
root.mainloop()
```

Evenimentele <ButtonRelease-1>, <ButtonRelease-3>

- Apelează funcția de manipulare a evenimentului atunci când se eliberează butonul stâng apăsat (<ButtonRelease-1>) sau butonul drept apăsat (<ButtonRelease-3>)

```
from tkinter import *
root=Tk()
root.geometry('400x400')

def stang_apasat(event):
    root.config(bg="red")
def drept_apasat(event):
    root.config(bg="yellow")
def stang_eliberat(event):
    root.config(bg="green")
def drept_eliberat(event):
    root.config(bg="blue")

buton=Button(root, text='Buton de manipulare')
buton.pack(expand = True)

buton.bind('<Button-1>', stang_apasat)
buton.bind('<Button-3>', drept_apasat)
buton.bind('<ButtonRelease-1>', stang_eliberat)
buton.bind('<ButtonRelease-3>', drept_eliberat)
root.mainloop()
```

Evenimentul <Motion>

- Apelează funcția de manipulare a evenimentului atunci când cursorul se mișcă pe componenta căreia este atribuit

```
from tkinter import *
root=Tk()
root.geometry('400x400')

def schimbare_culoare(event):
    text.pack_forget()
    if root.cget('bg')== "red":
        root.config(bg="blue")
    elif root.cget('bg')== "blue":
        root.config(bg="green")
    else:
        root.config(bg="red")
    text.config(text=f"Pozitie cursor ({event.x}, {event.y})")
    text.pack(expand = True)

text=Label(root)
root.bind('<Motion>', schimbare_culoare)
root.mainloop()
```

Evenimentul <Enter>

- Apelează funcția de manipulare a evenimentului atunci când cursorul intră pe suprafața componentei căreia este atribuit

```
from tkinter import *
from random import *

root=Tk()
root.geometry('600x600')

def prinde_ma(event):
    buton.place_forget()
    buton.place(x=randint(0,550), y=randint(0,570), width=50, height=30)

buton=Button(text="Click", bg='red', fg="white")

buton.place(x=275, y=285, width=50, height=30)

buton.bind('<Enter>', prinde_ma)

root.mainloop()
```

Evenimentul <Leave>

- Apelează funcția de manipulare a evenimentului atunci când cursorul părăsește suprafața componentei căreia este atribuit

```
from tkinter import *
root=Tk()
root.geometry('600x600')

def lasa_ma(event):
    root.update()
    x = buton.winfo_x()
    y = buton.winfo_y()
    buton.place_forget()
    buton.place(x=x+event.x-25, y=y+event.y-15, width=50, height=30)

buton=Button(text="Click", bg='red', fg="white")
buton.place(x=275, y=285, width=50, height=30)

buton.bind('<Leave>', lasa_ma)

root.mainloop()
```

Evenimentul < MouseWheel >

- Apelează funcția de manipulare a evenimentului atunci când se activează butonul rotativ al mouse-ului pe componentă

```
from tkinter import *  
root=Tk()  
root.geometry('400x400')  
  
def schimbare_culoare(event):  
    if root.cget('bg')== "red":  
        root.config(bg="blue")  
    elif root.cget('bg')== "blue":  
        root.config(bg="green")  
    elif root.cget('bg') == "green":  
        root.config(bg="yellow")  
    else:  
        root.config(bg="red")  
  
root.bind('<MouseWheel>', schimbare_culoare)  
  
root.mainloop()
```

Evenimentul < Double-Button >

- Apelează funcția de manipulare a evenimentului atunci când se apasă dublu click pe orice buton al mouse-ului pe componenta căreia este atribuit

```
from tkinter import *
root=Tk()
root.geometry('400x400')

def schimbare_culoare(event):
    if root.cget('bg')== "red":
        root.config(bg="blue")
    elif root.cget('bg')== "blue":
        root.config(bg="green")
    elif root.cget('bg') == "green":
        root.config(bg="yellow")
    else:
        root.config(bg="red")

root.bind('<Double-Button>', schimbare_culoare)

root.mainloop()
```


3. Evenimente asociate tastaturii

Evenimentul <Key>

- Evenimentele asociate tastaturii pot fi atribuite doar ferestrei de bază
- Evenimentul <Key> apelează funcția de manipulare a evenimentului atunci când se apasă orice buton al tastaturii

```
from tkinter import *  
root=Tk()  
root.geometry('400x400')
```

```
def apasare_buton(event):  
    text.pack_forget()  
    text.pack(fill = BOTH, expand = True)  
    if text.cget('bg')== "gold":  
        text.config(bg="silver")  
    else:  
        text.config(bg="gold")
```

```
text=Label(text="Ai apasat un buton al tastaturii", font="Calibri 16 bold")  
root.bind('<Key>', apasare_buton)
```

```
root.mainloop()
```

Atributele obiectului evenimentului <Key>

- Evenimentul <Button> creează un obiect al evenimentului care este transmis în mod automat funcției de manipulare a acestuia. Acest obiect are un atribut: `keysym` care specifică ce buton al tastaturii a fost apăsată

```
from tkinter import *
root=Tk()
root.geometry('400x400')

def apasare_buton(event):
    text.pack_forget()
    text.pack(fill = BOTH, expand = True)
    text.config(text=f"Ai tastat butonul {event.keysym}")
    if text.cget('bg')=="red":
        text.config(bg="green")
    else:
        text.config(bg="red")

text=Label(text="A apasat un buton al tastaturii", fg="white", font="Calibri 16 bold")
root.bind('<Key>', apasare_buton)
root.mainloop()
```

Evenimentele <KeyPress> și <KeyRelease>

- Apelează funcția de manipulare a evenimentului atunci când se apasă un buton al tastaturii (<KeyPress>) sau se eliberează un buton apăsăat al tastaturii (<KeyRelease>)

```
from tkinter import *
root=Tk()
root.geometry('400x400')

def apasare_buton(event):
    text.pack_forget()
    text.pack(fill = BOTH, expand = True)
    text.config(text=f"Ai apasat pe butonul {event.keysym}")
    text.config(bg="red")
def eliberare_buton(event):
    text.pack_forget()
    text.pack(fill = BOTH, expand = True)
    text.config(text=f"Ai eliberat butonul {event.keysym}")
    text.config(bg="green")

text=Label(fg="white", font="Calibri 16 bold")

root.bind('<KeyPress>', apasare_buton)
root.bind('<KeyRelease>', eliberare_buton)
root.mainloop()
```

Evenimente asociate tastelor speciale

- Evenimentele pot fi asociate separat fiecărui buton al tastaturii
- Pentru crearea evenimentelor butoanelor speciale se include denumirea lor între semnele <>
- Butoanele speciale se consideră butoanele: F1, F2, ..., F12, Left, Right, Up, Down, BackSpace, Delete, End, Home, Insert, Print, Tab
- Excepție fac butoanele Enter cu evenimentul <Return> și Esc cu evenimentul <Escape>

```
from tkinter import *  
root=Tk()  
root.geometry('600x600')
```

```
def dreapta(event):  
    root.update()  
    x=buton.wininfo_x()  
    y=buton.wininfo_y()  
    buton.place_forget()  
    buton.place(x=x+1, y=y, width=50, height=30)
```

```
def sus(event):  
    root.update()  
    x=buton.wininfo_x()  
    y=buton.wininfo_y()  
    buton.place_forget()  
    buton.place(x=x, y=y-1, width=50, height=30)
```

Evenimente asociate tastelor speciale (continuare)

```
def stanga(event):
    root.update()
    x=buton.winfo_x()
    y=buton.winfo_y()
    buton.place_forget()
    buton.place(x=x-1, y=y, width=50, height=30)

def jos(event):
    root.update()
    x=buton.winfo_x()
    y=buton.winfo_y()
    buton.place_forget()
    buton.place(x=x, y=y+1, width=50, height=30)

def centru(event):
    buton.place_forget()
    buton.place(x=275, y=285, width=50, height=30)

buton=Button(root, text="Click", bg='red', fg="white")
buton.place(x=275, y=285, width=50, height=30)
root.bind('<Right>', dreapta)
root.bind('<Up>', sus)
root.bind('<Left>', stanga)
root.bind('<Down>', jos)
root.bind('<Escape>', centru)
root.mainloop()
```

Evenimente asociate tastelor normale

- Pentru crearea evenimentelor butoanelor normale se trece denumirea lor fără semnele <>
- Exemplu de evenimente a butoanelor normale: a, A, +

```
from tkinter import *  
root=Tk()  
root.geometry('400x400')
```

```
def a_mic(event):  
    text.pack_forget()  
    root.config(bg="red")  
    text.config(text="Ai tastat simbolul a")  
    text.pack(expand=True)
```

```
def a_mare(event):  
    text.pack_forget()  
    root.config(bg="yellow")  
    text.config(text="Ai tastat simbolul A")  
    text.pack(expand=True)
```

Evenimente asociate tastelor normale (continuare)

```
def plus(event):  
    text.pack_forget()  
    root.config(bg="green")  
    text.config(text="Ai tastat simbolul +")  
    text.pack(expand=True)  
  
def bara(event):  
    text.pack_forget()  
    root.config(bg="blue")  
    text.config(text="Ai tastat simbolul /")  
    text.pack(expand=True)  
  
text=Label(root, font="Calibri 16 bold")  
  
root.bind('a', a_mic)  
root.bind('A', a_mare)  
root.bind('+', plus)  
root.bind('/', bara)  
  
root.mainloop()
```

Evenimente asociate combinațiilor de taste

- Pentru crearea evenimentelor combinațiilor de butoanelor se notează denumirea tastelor respective separate prin simbolul - incluse între semnele <>
- Exemplu de evenimente a combinațiilor de butoane: <Alt-a>, <Control-Up>
- Exemplu cu completarea programului de la punctul: Evenimente asociate tastelor speciale

...

```
def dreapta_rapid(event):  
    root.update()  
    x=buton.winfo_x()  
    y=buton.winfo_y()  
    buton.place_forget()  
    buton.place(x=x+10, y=y, width=50, height=30)
```

```
def sus_rapid(event):  
    root.update()  
    x=buton.winfo_x()  
    y=buton.winfo_y()  
    buton.place_forget()  
    buton.place(x=x, y=y-10, width=50, height=30)
```


Evenimente asociate combinațiilor de taste (continuare)

```
def stanga_rapid(event):  
    root.update()  
    x=buton.winfo_x()  
    y=buton.winfo_y()  
    buton.place_forget()  
    buton.place(x=x-10, y=y, width=50, height=30)  
  
def jos_rapid(event):  
    root.update()  
    x=buton.winfo_x()  
    y=buton.winfo_y()  
    buton.place_forget()  
    buton.place(x=x, y=y+10, width=50, height=30)  
  
...  
root.bind('<Control-Right>', dreapta_rapid)  
root.bind('<Control-Up>', sus_rapid)  
root.bind('<Control-Left>', stanga_rapid)  
root.bind('<Control-Down>', jos_rapid)  
...
```