

Bölüm 7

Radiobutton

7.1 Radyo Düğmesi nedir?

Radiobutton (radyo düğmesi), tkinter içinde bir pencere aracıdır (widget). Görünümü, eski tip radyoların düğmelerine benzer. İşlevi *Checkbox* (seçim düğmesi) gibidir. Genellikle birden çok *radiobutton* bir arada olur. Bu grup içinden seçim yapılır. *Checkbox* grubundan istenildiği kadar seçim yapılabilir. Ama *radiobuttons* grubundan kullanıcı ancak bir tanesini seçebilir.

Radiobutton nesnesi arayüzde beyaz renkte oyuk bir disk gibi görünür. Seçili olunca diskin ortasına siyah bir nokta yerleşir.

Radyo düğmesi, düğmenin ne yaptığını belirten yazı ya da resim içerir. Yazılar birer *label* olduğundan tek tür font ile yazılır. *Label*'de bir harfi alt çizgili olabilir. Radyo düğmesi bir Python fonksiyonuna bağlanabilir. Başka bir deyişle, düğme seçildiğinde kendisine bağlı fonksiyon işler. *Radiobutton* düğmesi, *checkbox* için olduğu gibi seçili/seçisiz (on/off) olmak üzere iki değer alabilir. Seçisiz iken tıklayınca seçili olur, seçii iken tıklayınca seçisiz olur. Ayrıca, bütün grup içinden birisi seçilince ötekiler seçisiz olur.

Liste 7.1.

```
from tkinter import *  
  
4 pencere = Tk()  
v = IntVar()  
  
9 Label(pencere,  
    text="Birini seç",  
    justify = LEFT,
```

```

    padx = 20).pack()
Radiobutton(pencere,
            text="Java",
            padx = 20,
14         variable=v,
            value=1).pack(anchor=W)
Radiobutton(pencere,
            text="Python",
            padx = 20,
19         variable=v,
            value=2).pack(anchor=W)
mainloop()

```

Açıklamalar:

1., 3. ve 22. satırların işlevini biliyoruz.

5.satır *IntVar* adlı bir fonksiyon bildirimidir. Fonksiyona *v* adı verilmiştir. *tkinter*'de sayısal değer alan fonksiyonu *IntVar*, string değer alan fonksiyonu *StringVar* diye adlandırırız. Böyle yapmakla, algıyı kolaylaştırmış oluruz.

7.satır bir *Label* nesnesi yaratıyor. İlk parametresi *pencere* olduğuna göre, *Label* nesnesi pencere (anataşıcı) üzerine konulacaktır.

8.satır, etiket olarak yazılacak yazıyı *text* parametresi ile belirliyor.

9.satır, label üzerindeki yazının, label'in soluna yanaşık olacağını belirtiyor.

10.satır, yazı ile *label*'in kenarları arasında 20 pixel genişliğinde yas-tıklama yapıyor. Son olrak *Label().pack()* ifadesi, *pack()* konuştur-cısını *Label* nesnesine uyguluyor. *pack()* konuşturıcısı parametresiz ol-duğu için, öndegerleri çalışıyor ve *Label* nesnesini pencerenin kuzey kenarına yanaşık koyuyor ve pencere genişliğine göre ort alıyor.

11.satır, bir *Radiobutton* nesnesi yaratıyor. İlk parametresi *pencere* olduğuna için, yaratılan *radiobutton* pencere üzerine konulacaktır.

12. ve 13. satırların işlevi açıktır.

14.satır, *variable* değişkenine *IntVar()* fonksiyonunun değerini atıyor.

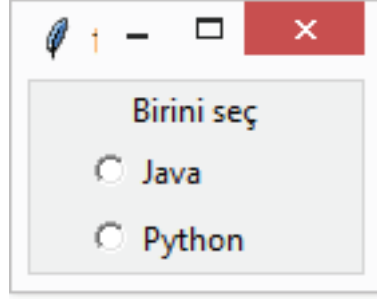
15.satır, *value* parametresine 2 değerini atıyor. Son olarak

```
| \texttt{Radiobutton().pack(anchor=W)}
```

ifadesi, yaratılan *radiobutton* düğmesini *pack()* ile pencere üzerine ko-nuşturuyor. *anchor=W* parametresi, *radiobutton* nesnesini pencerenin *W* (batı) kenarına yanaşık konuşturuyor.

16-20. satırlar ikinci bir *radiobutton* yaratıyor ve birinciye benzer işleri yapıyorlar.

Ekranında oluşan arayüzü enine/boyuna genişletirseniz, Label'in pencerenin N (kuzey) kenarına yanaşık ve genişliğe göre ortalı olduğunu görebilirsiniz. Benzer olarak, yaratılan iki düğme pencerenin W (batı) kenarına yanaşık olarak konuşlanmışlardır. Pencere büyüklüğünün değişmesi, onların konumlarını değiştirmez.



Şekil 7.1: İkili Radiobutton

7.1.1 Çoklu Düğmeler

Uygulamada çok sayıda radiobutton bir grup oluşturur. Her düğmeyi, yukarıdaki gibi tek tek yaratmak mümkündür. Ama Python diline dayalı iş yaptığımızı göre, ilkemiz bir işi tekrar etmekten sakınmak olmalıdır. Radiobutton düğmeleri benzer olacağına göre, onların adlarını bir `list` ambarına koyup, oradan çağırabiliriz. Sonra bir döngü ile radiobutton nesnelerini yaratabiliriz. Module 7.1 o işi yapıyor.

Module 7.1.

```

from tkinter import *
root = Tk()
4 v = IntVar()
v.set(1) # seçim için ilk değeri veriyor

diller = [
9     ("Python", 1),
    ("Ruby", 2),
    ("Perl", 3),
    ("Java", 4),
14    ("C++", 5)
]

def SeçimiGöster():
    print(v.get())

```

```

19 Label(root ,
      text="Favori dilinizi seçin" ,
      justify = LEFT,
      padx = 20).pack()
24 for txt, val in diller:
      Radiobutton(root ,
                  text=txt ,
                  padx = 20,
                  variable=v ,
29          command=SeçimiGöster ,
                  value=val).pack(anchor=W)

mainloop()

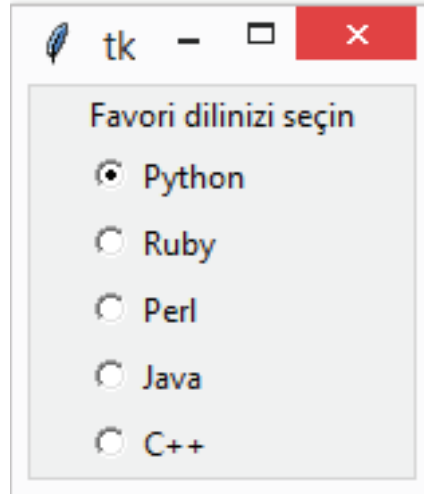
```

Açıklamalar:

8.satır, seçenekleri bir list ambarına koyuyor.

24.satır, bir döngü ile list ambarındaki her öğe için bir radiobutton nesnesi yaratıyor.

29.satır, her düğmeye `SeçimiGöster` fonksiyonunu bağlıyor. Bir düğme seçilince, 1,2,3,4,5 sayısal değerlerinden ilgili olan radiobutton düğmesi görüntüye geliyor.



Şekil 7.2: Döngü ile Radiobutton Yaratma

7.2 Indicator

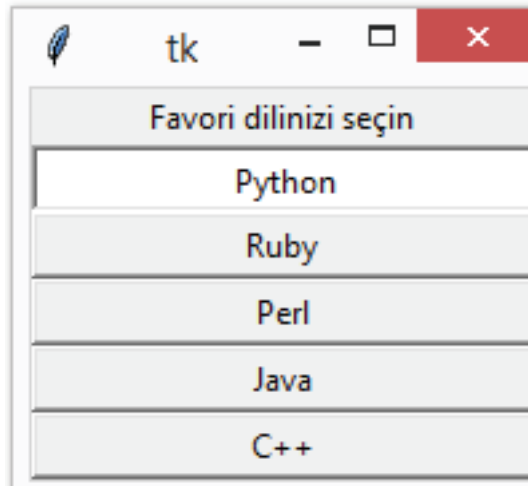
Bazen, estetik nedenlerle, yuvarlak disk görünümündeki radiobutton yerine, işlevini gösterir etiketli birer kutu olarak görünmesini sağlayabiliriz. Bunun için Radiobutton nesnesini yaratan kurucuda indicatoron = 0 parametresini kullanmak yetercektir. Module 7.1'de 21. satırdan sonrakiler için şunları yazarak, programı yeniden koşturunuz.

```

Radiobutton(root ,
            text=txt ,
3           indicatoron = 0 ,
            width = 20 ,
            padx = 20 ,
            variable=v ,
8           command=ShowChoice ,
            value=val) . pack ( anchor=W)

```

Ekrana gelen arayüz şöyle olacaktır:



Şekil 7.3: Indicatoron Parametrelili

Module 7.2.

```

from tkinter import *
2 root = Tk()
  root.title( 'Radiobutton ' )

dersler=[( 'Biyoloji' , 1) , ( 'Fizik' , 2) , ( 'Kimya' , 3) ,
7         ( 'Matematik' , 4)]
var = IntVar()
for text , value in dersler :

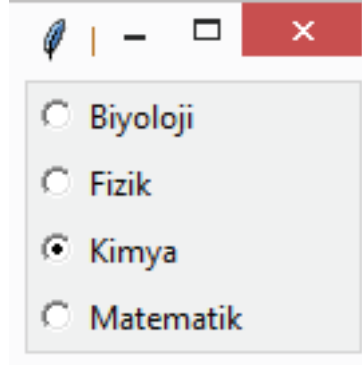
```

```

    Radiobutton(root, text=text, value=value, variable=var).pack(
        anchor=W)
var.set(3)
12 root.mainloop()

```

Açıklamalar:



Şekil 7.4: Dersler

Module 7.3.

```

from Tkinter import *
2 root = Tk()
  root.title('Radiobutton')
cheese=[('Option 1', 1), ('Option 2', 2), ('Option 3', 3),
7      ('Option 4', 4), ('Option 5', 5), ('Option 6', 6), ('Option 7',
      7)]
var = IntVar()
for text, value in cheese:
    Radiobutton(root, text=text, value=value, variable=var,
12      indicatoron=0).pack(anchor=W, fill=X, ipadx=18)
var.set(3)
root.mainloop()

```

Açıklamalar:

Module 7.4.

```

from Tkinter import *
2 class AllTkinterWidgets:
    def __init__(self, master):
        frame = Frame(master, width=500, height=400, bd=1)
        frame.pack()
7

```

```

    iframe1 = Frame(frame, bd=2, relief=SUNKEN)
    Button(iframe1, text='Button').pack(side=LEFT, padx=5)
    Checkbutton(iframe1, text='CheckButton').pack(side=LEFT, padx
=5)
12     v=IntVar()
    Radiobutton(iframe1, text='Button', variable=v,
                value=3).pack(side=RIGHT, anchor=W)
    Radiobutton(iframe1, text='Dio', variable=v,
                value=2).pack(side=RIGHT, anchor=W)
17     Radiobutton(iframe1, text='Ra', variable=v,
                value=1).pack(side=RIGHT, anchor=W)
    iframe1.pack(expand=1, fill=X, pady=10, padx=5)

22 root = Tk()
#root.option_add('*font', ('verdana', 10, 'bold'))
all = AllTkinterWidgets(root)
root.title('Tkinter Widgets')
27 root.mainloop()

```

Açıklamalar:

Module 7.5.

```

from Tkinter import *
3 root = Tk()

var = IntVar()

for i in range(10):
8     rad = Radiobutton(root, text=str(i), value=i, variable=var)
        rad.pack(side=LEFT)

root.mainloop()
print var.get()

```

Açıklamalar:

Module 7.6.

```

from Tkinter import *
3 root = Tk()

def radiol():
    tmp = IntVar()
    for i in range(10):
8         rad = Radiobutton(root, text=str(i), value=i, variable=tmp)
            rad.pack(side=LEFT)
        tmp.set(5)

radiol()
root.mainloop()

```

Açıklamalar:

Module 7.7.

```
from Tkinter import *
state = ''
3 buttons = []

def onPress(i):
    global state
    state = i
8     for btn in buttons:
        btn.deselect()
        buttons[i].select()

root = Tk()
13 for i in range(10):
    rad = Radiobutton(root,
                      text=str(i),
                      value=str(i),
                      command=(lambda i=i: onPress(i)) )
18     rad.pack(side=LEFT)
    buttons.append(rad)
root.mainloop()

print state
```

Açıklamalar: