

## *Python - GUI Programlama (Tkinter)*

### **1. Tkinter Nedir?**

Tkinter, Python için standart GUI(Graphical User Interface – Grafiksel Kullanıcı Arayüzü) kütüphanesidir. Python, Tkinter ile birlikte kullanıldığında GUI uygulamaları hızlı ve kolay bir şekilde oluşturulur. Tkinter, Tk GUI araç setine güçlü bir nesne yönelimli arayüz sağlar. Python2’de “Tkinter”, Python3’te “tkinter” modülü ile temsil edilir.

Tkinter kullanarak bir GUI uygulaması oluşturmak kolaydır. Tek yapmamız gereken aşağıdaki adımları gerçekleştirmektir;

- Tkinter modülünü uygulamaya import edin.
- GUI uygulama ana penceresini oluşturun.
- GUI uygulamasına yukarıda belirtilen widget'lardan birini veya daha fazlasını ekleyin.
- Kullanıcı tarafından tetiklenen her etkinliğe karşı harekete geçmek için ana etkinlik döngüsünü girin.

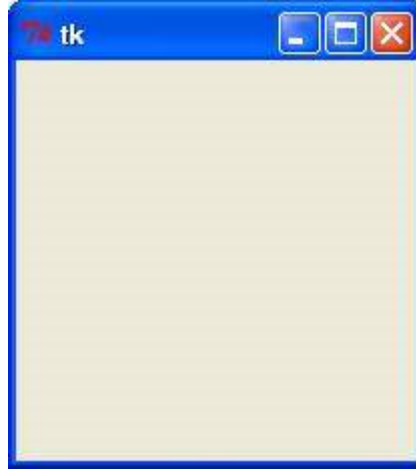
Örnek – 1 :

```
import tkinter as tk      #python3 için
mainFrame = tk.Tk()

# Widget eklemek için gerekli kodu buraya yazalım

mainFrame.mainloop()
```

Bu kod aşağıdaki pencereyi oluşturur



Şekil-1

## 2. Tkinter Widget'ları Nelerdir?

- **Button**  
Uygulamanızda butonları görüntülemek için kullanılır.
- **Canvas**  
Uygulamanızda çizgiler, ovaler, çokgenler ve dikdörtgenler gibi şekiller çizmek için kullanılır.
- **Checkbutton**  
Onay kutularını bir dizi seçenek olarak görüntülemek için kullanılır. Kullanıcı bir seferde birden fazla seçenek seçebilir.
- **Entry**  
Kullanıcıdan değerleri kabul etmek için tek satırlık bir metin alanını görüntülemek için kullanılır.
- **Frame**  
Diğer widget'ları düzenlemek için bir konteyner widget'ı olarak kullanılır.
- **Label**  
Diğer widget'lar için tek satırlık yazı sağlamak için kullanılır. Ayrıca görüntüler de içerebilir.
- **Listbox**  
Kullanıcıya seçeneklerin listesini sağlamak için kullanılır.
- **Menubutton**  
Uygulamanızdaki menüleri görüntülemek için kullanılır.
- **Menu**  
Kullanıcıya çeşitli komutlar sağlamak için kullanılır. Bu komutlar Menubutton içinde yer almaktadır.
- **Message**  
Kullanıcıdan değerleri kabul etmek için çok satırlı metin alanlarını görüntülemek için kullanılır.
- **Radiobutton**  
Radyo düğmeleri olarak bir dizi seçeneği görüntülemek için kullanılır. Kullanıcı bir seferde sadece bir seçenek seçebilir.
- **Scale**  
Bir kaydırıcı widget'ı sağlamak için kullanılır.
- **Scrollbar**  
Liste kutuları gibi çeşitli widget'lara kaydırma özelliği eklemek için kullanılır.
- **Text**  
Metni birden çok satırda görüntülemek için kullanılır.
- **Toplevel**  
Ayrı bir pencere kabı sağlamak için kullanılır.
- **Spinbox**  
Sabit sayıda sayıdan seçim yapmak için kullanılabilen standart Tkinter Giriş widget'ının bir çeşididir.
- **PanedWindow**  
Yatay veya dikey olarak düzenlenmiş herhangi bir sayıda bölmeyi içerebilen bir kapsayıcı pencere ögesidir.
- **LabelFrame**  
Basit konteyner widget'ıdır. Karmaşık pencere düzenleri için bir konteyner olarak kullanılır.
- **messagebox**  
Bu modül, uygulamalarınızda mesaj kutularını görüntülemek için kullanılır.

### 3. Standart Öznitelikler

- **Dimensions**  
Widget'ların çeşitli uzunlukları, genişlikleri ve diğer boyutları farklı birimlerle tanımlanabilir.
- **Colors**  
Renkleri temsil eder.
- **Fonts**  
Yazı stilini temsil eder.
- **Anchors**  
Metnin bir referans noktasına göre nerede konumlandırılacağını tanımlamak için kullanılır.
- **Relief styles**  
Widget'ın kabartma stili veya widget'ın dışındaki bazı simüle 3-D efektleri anlamına gelir.
- **Bitmaps**  
Bu özellik gösterilecek bitmap'i (icon'u) temsil eder.
- **Cursors**  
Fare imlecini temsil eder. Fare imlecini bu özellik sayesinde ayarlanır.

### 4. Tkinter Widget'larının Detaylı İncelenmesi

- **Button**  
Buton eklemek için kullanılır. Bu butonlar ile butonun yapacağı işi temsil eden metin veya resim görüntülenebilir. Butona tıkladığınızda otomatik olarak çağırılan fonksiyon veya metod ekleyebilirsiniz.

#### Sözdizimi (Syntax)

```
w = Button( master, option=value, ... )
```

#### Parametreler

- **master** : Ana pencereyi temsil eder
- **options** : Seçenekler listesini temsil eder. Bu seçenekler virgülle ayrılmış anahtar – değer çiftleri olarak kullanılır. En çok kullanılan seçeneklerin listesi aşağıda belirtilmiştir.

Sr.No.	Seçenek & Açıklama
1	<b>Activebackground</b> Mouse ile butonun üzerine tıklandığındaki butonun arka plan rengi.
2	<b>Activeforeground</b> Mouse ile butonun üzerine tıklandığındaki butonun yazı rengi.
3	<b>Bd</b> Piksel cinsinden kenarlık genişliği. Varsayılan 2.

4	<b>Bg</b> Normal arka plan rengi.
5	<b>command</b> Butona tıklandığında çağırılacak fonksiyon yada metod adı.
6	<b>fg</b> Normal yazı rengi.
7	<b>font</b> Buton etiketi için kullanılacak metin yazı tipi.
8	<b>height</b> Yükseklik.
9	<b>highlightcolor</b> Fokus olunduğunda ki rengi.
10	<b>image</b> Üzerinde metin yerine görüntülenecek resim.
11	<b>justify</b> Text birden çok satır içerirse, text'in nereye yanaşık olacağını belirler. CENTER değeri ön değerdir. LEFT ve RIGHT değerleri de verilebilir.
12	<b>padx</b> Metnin sol ve sağ ek dolguları.
13	<b>pady</b> Metnin üstünde ve altında ek dolguları.
14	<b>relief</b> Butonun kenar (border) tipini belirtir. Bazı değerler SUNKEN, RAISED, GROOVE ve RIDGE.
15	<b>state</b> Durumu temsil eder. DISABLED durumu pasif eder, NORMAL ise durumu ise aktif eder. Varsayılan durum NORMAL'dir.

16	<p><b>Underline</b></p> <p>Varsayılan değeri -1'dir. Buton metninin hiçbir karakterinin altının çizilmeyeceğini belirtir. Eğer bu değer negatif değil ise karşılık gelen metin karakterinin altı çizilir.</p>
17	<p><b>width</b></p> <p>Buton içindeki harflerin genişliği (metin görüntüleniyorsa) veya pikseller (bir görüntü gösteriliyorsa).</p>
18	<p><b>wraplength</b></p> <p>Bu değer pozitif bir sayıya ayarlanırsa, metin satırları bu uzunluğa sığacak şekilde ayarlanır.</p>

### Metodlar

Button için en yaygın kullanılan metodlar şunlardır;

Sr.No.	Metod & Açıklama
1	<p><b>flash()</b></p> <p>Düğmenin aktif ve normal renkler arasında birkaç kez yanıp sönmesine neden olur. Düğmeyi başlangıçta olduğu durumda bırakır. Düğme devre dışıysa göz ardı edilir.</p>
2	<p><b>invoke()</b></p> <p>Düğmenin geri aramasını çağırır ve bu işlevin geri döndüğünü döndürür. Düğme devre dışı bırakılmışsa veya geri arama yoksa etkisi yoktur.</p>

Button kullanımına ait örnek kod Şekil – 2’de ve kodun çıktısı da Şekil – 2.1’de gösterilmiştir.

```
In [115]: import tkinter as tk
from tkinter import messagebox
from PIL import Image, ImageTk

mainFrame = tk.Tk()

def messageCallBack() :
    messagebox.showinfo("Dikkat!", "Bu bir mesaj")

def imageCallBack() :
    messagebox.showinfo("Dikkat!", "Bu bir image")

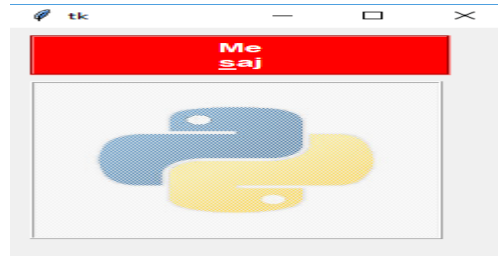
image = Image.open("python.png")
photo = ImageTk.PhotoImage(image)

btnMessage = tk.Button(text = "Mesaj",
    activebackground = "white",
    activeforeground = "red",
    bd = 2,
    bg = "red",
    fg = "white",
    command = messageCallBack,
    font = "Arial 12 bold",
    height = 2,
    width = 20,
    highlightcolor = "green",
    justify = "center",
    relief = "groove",
    state = "normal",
    underline = 2,
    wraplength = 30)
btnMessage.place(x = 10, y = 10)

btnImage = tk.Button(image = photo,
    command = imageCallBack,
    width = 200, #Pixel
    relief = "ridge",
    state = "disabled")
btnImage.place(x = 10, y = 70)

mainFrame.mainloop()
```

Şekil – 2



Şekil – 2.1

➤ **Canvas**

Çizgiler, ovaler, çokgenler ve dikdörtgenler gibi şekiller çizmek için kullanılır.

Sözdizimi (Syntax)

```
w = Canvas( master, option=value, ... )
```

Parametreler

- **master** : Ana pencereyi temsil eder
- **options** : Seçenekler listesini temsil eder. Bu seçenekler virgülle ayrılmış anahtar – değer çiftleri olarak kullanılır. En çok kullanılan seçeneklerin listesi aşağıda belirtilmiştir.

Sr.No.	Seçenek & Açıklama
1	<b>bd</b> Piksel cinsinden kenarlık genişliği. Varsayılan 2'dir.
2	<b>bg</b> Normal arkaplan rengi.
3	<b>Confine</b> True (varsayılan) ise, canvas kaydırma bölgesinin dışına kaydırılamaz.
4	<b>cursor</b> İmleç tuvalde ok, daire, nokta vb gibi kullanılır.
5	<b>height</b> Y boyutundaki tuvalin boyutu. Yükseklik.
6	<b>highlightcolor</b> Odak vurgulamasında gösterilen renk.
7	<b>relief</b> Kenarlık türünü belirtir. Bazı değerler SUNKEN, RAISED, GROOVE ve RIDGE'dir.

8	<p><b>scrollregion</b></p> <p>Bir tuvalin ne kadar büyük bir alana kaydırılacağını tanımlayan bir tuple (w, n, e, s). <b>w</b> sol taraf, <b>n</b> üst taraf, <b>e</b> sağ taraf ve <b>s</b> alt kısımdır.</p>
9	<p><b>width</b></p> <p>X boyutunda tuvalin boyutu.</p>
10	<p><b>xscrollincrement</b></p> <p>Bu seçeneği bazı pozitif boyutlara ayarlarsanız, kanvas yalnızca bu mesafenin katları üzerine konumlandırılabilir ve değer kaydırma çubuğuyla kaydırma yapmak için kullanılır, örneğin kullanıcının kaydırma çubuğunun uçlarındaki okları tıkladığında olduğu gibi.</p>
11	<p><b>xscrollcommand</b></p> <p>Tuval kaydırılabilirse, bu özellik yatay kaydırma çubuğunun .set () yöntemi olmalıdır.</p>
12	<p><b>yscrollincrement</b></p> <p>xscrollincrement gibi çalışır, ancak dikey hareketi yönetir.</p>
13	<p><b>yscrollcommand</b></p> <p>Tuval kaydırılabilirse, bu özellik dikey kaydırma çubuğunun .set () yöntemi olmalıdır.</p>

Tuval widget'ı aşağıdaki standart öğeleri destekleyebilir –

**arc** – Canvas üzerine çizilen elips ya da çemberden yaratılan parça. Özel olarak elips ya da çemberin tamamı da birer arc nesnesi sayılır.

```
coord = 10, 50, 240, 210
arc = canvas.create_arc(coord, start=0, extent=150, fill="blue")
```

**image** – Grafik çizer. BitmapImage veya PhotoImage sınıflarının bir örneği olabilecek bir görüntü öğesi oluşturur.

```
filename = PhotoImage(file = "sunshine.gif")
image = canvas.create_image(50, 50, anchor=NE, image=filename)
```



**line** – Doğru parçası çizer.

```
line = canvas.create_line(x0, y0, x1, y1, ..., xn, yn, options)
```

**oval** – Verilen koordinatlarda bir daire veya elips çizer. İki çift koordinat alır; oval için sınırlayıcı dikdörtgenin sol üst ve alt sağ köşeleri.

```
oval = canvas.create_oval(x0, y0, x1, y1, options)
```

**polygon** – En az üç köşeye sahip olması gereken çokgen bir öge çizer.

```
oval = canvas.create_polygon(x0, y0, x1, y1,...xn, yn, options)
```

Canvas kullanımına ait örnek kod Şekil – 3’de ve kodun çıktısı da Şekil – 3.1’de gösterilmiştir.

```
In [44]: import tkinter as tk

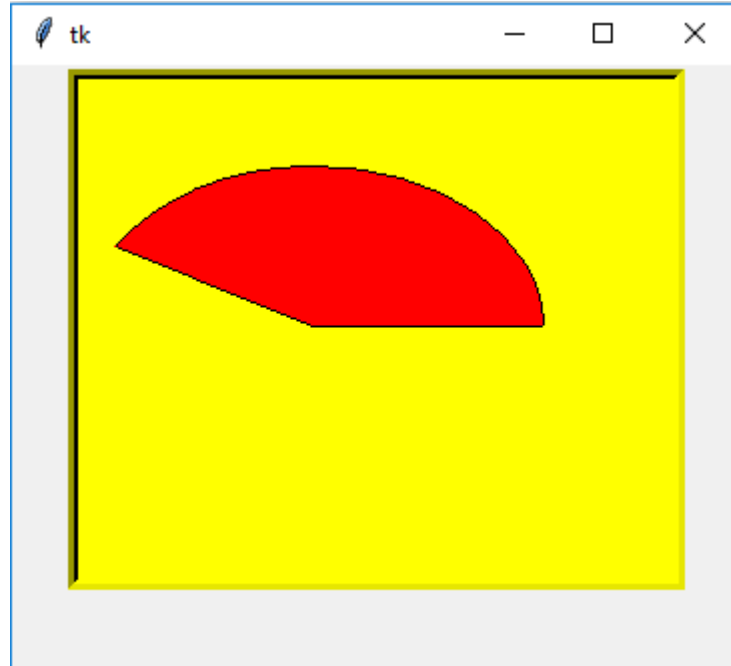
mainFrame = tk.Tk()

mycanvas = tk.Canvas(mainFrame,
                      bd = 5,
                      bg = "yellow",
                      confine = True,
                      cursor = "mouse",
                      height = 250,
                      highlightcolor = "red",
                      relief = "sunken",
                      width = 300)

coord = 10, 50, 240, 210
arc = mycanvas.create_arc(coord, start=0, extent=150, fill="red")

mycanvas.pack()
mainFrame.mainloop()
```

Şekil – 3



Şekil – 3.1

### ➤ Checkbutton

Onay kutularını bir dizi seçenek olarak görüntülemek için kullanılır. Kullanıcı bir seferde birden fazla seçenek seçebilir.

#### Sözdizimi (Syntax)

```
w = Checkbutton( master, option=value, ... )
```

#### Parametreler

- **master** : Ana pencereyi temsil eder
- **options** : Seçenekler listesini temsil eder. Bu seçenekler virgülle ayrılmış anahtar – değer çiftleri olarak kullanılır. En çok kullanılan seçeneklerin listesi aşağıda belirtilmiştir.

Sr.No.	Seenek & Açıklama
1	<b>activebackground</b> Onay düğmesi imlecin altında olduğunda arka plan rengi.
2	<b>activeforeground</b> Onay düğmesi imlecin altında olduğunda ön plan rengi.
3	<b>bg</b> Normal arka plan rengi. Örneğın, bg="red" seçilirse düğme kırmızı olur.
4	<b>bitmap</b> Tek renkli bir görüntüyü bir düğmede görüntülemek için.
5	<b>bd</b> Gösterge etrafındaki sınırın boyutu. Varsayılan 2 pikseldir.
6	<b>command</b> Kullanıcı bu kontrol düğmesinin durumunu her değıştirdiğinde çağrılacak fonksiyon.
7	<b>cursor</b> Bu seçeneğı bir imleç ismine ( <i>arrow, dot</i> vb.) ayarlarsanız, fare imleci, kontrol

	tuşunun üzerindeyken o desene değişir.
8	<b>disabledforeground</b> Devre dışı bırakılan bir denetimin metnini oluşturmak için kullanılan önalan rengi. Varsayılan, varsayılan ön plan renginin noktalı bir sürümüdür.
9	<b>font</b> Metin için kullanılan yazı tipi.
10	<b>fg</b> Metni oluşturmak için kullanılan renk.
11	<b>height</b> Kontrol düğmesindeki metin satırlarının sayısı. Varsayılan 1'dir.
12	<b>highlightcolor</b> Fare ile imleç taşıyıcının düğmeleri üzerine gittiğinde, düğmenin alacağı aydınlanma rengini belirler.
13	<b>image</b> Widget üzerine konulacak resim. Bitmap ve text'e göre resim öncelik alır.
14	<b>justify</b> Widget üzerine birden çok satır yazılınca, text'in nereye yanaşık olacağını belirler. CENTER değeri ön değerdir. İstenirse LEFT ve RIGHT değerleri de verilebilir.
15	<b>offvalue</b> Checkbutton seçili değil iken aldığı değerdir. Ön değeri 0'dır.
16	<b>onvalue</b> Checkbutton seçili iken aldığı değerdir. Ön değeri 1'dir.

17	<p><b>padx</b></p> <p>Checkbox ve text'in soluna ve sağına ne kadar boşluk bırakılacağını belirtir. Ön değeri 1 pikseldir.</p>
18	<p><b>pady</b></p> <p>Checkbox ve text'in üstünde ve altında ne kadar boşluk bırakılacağını belirtir. Ön değeri 1 pikseldir.</p>
19	<p><b>relief</b></p> <p>Varsayılan değeri olan relief = FLAT ile, kontrol düğmesi arka planında göze çarpmaz. Bu seçeneği diğer stillerden birine ayarlayabilirsiniz.</p>
20	<p><b>Selectcolor</b></p> <p>Ayarlandığında checkbox'un rengi. Ön tanımlı selectcolor = "red".</p>
21	<p><b>selectimage</b></p> <p>Bu seçeneği bir görüntüye ayarlarsanız, bu görüntü checkbox üzerinde görünecektir.</p>
22	<p><b>state</b></p> <p>Varsayılan state=NORMAL durumundadır, ancak kontrolü bozmak ve yanıt vermemesi için state=DISABLED ögesini kullanabilirsiniz. İmleç checkbox'un üzerinde ise durum AKTİF.</p>
23	<p><b>text</b></p> <p>Checkbox yanında görünen etiket. Birden fazla metin satırı görüntülemek için yeni satırları ("\\ n") ile kullanın.</p>
24	<p><b>underline</b></p> <p>Varsayılan -1 değeriyle, metin etiketinin hiçbir karakterinin altı çizilemez. Bu seçeneği, karakterin altını çizmek için metindeki bir karakterin dizinine (sıfırdan sayma) ayarlayın.</p>

25	<p><b>variable</b></p> <p>Checkbox'un mevcut durumunu tutan deęişken. Normalde bu deęişken bir IntVar'dır. 0 seçilmemiş ve 1 seçilmiş demektir. Ancak yukarıdaki offvalue ve onvalue seçeneklerine bakınız.</p>
26	<p><b>width</b></p> <p>Bir kontrol düğmesinin varsayılan genişliği, görüntülenen resmin veya metnin boyutuna göre belirlenir. Bu seçeneęi bir dizi karaktere ayarlayabilir ve kontrol panelinde her zaman bu kadar çok karakter için yer olacaktır.</p>
27	<p><b>wraplength</b></p> <p>Normalde satırlar kaydırılmaz. Bu seçeneęi bir dizi karaktere ayarlayabilir ve tüm satırlar bu sayıdan daha uzun olmayan parçalara ayrılacaktır.</p>

#### Metodlar

Checkbox için en yaygın kullanılan metodlar şunlardır;

Sr.No.	Metod & Açıklama
1	<p><b>deselect()</b></p> <p>Checkbox'u temizler (kapatır).</p>
2	<p><b>flash()</b></p> <p>Checkbox'u aktif ve normal renkleri arasında birkaç kez yanıp söner, ancak başlattığı şekilde bırakır.</p>
3	<p><b>invoke()</b></p> <p>Kullanıcı, durumu deęiştirmek için checkbox'a tıkladığında oluşacak aynı eylemleri almak için bu yöntemi çağırabilirsiniz.</p>
4	<p><b>select()</b></p> <p>Checkbox'u ayarlar (seçer).</p>

5

**toggle()**

Checkbutton ayarlı ise temizler, temiz ise ayarlar.

Checkbutton kullanımına ait örnek kod Şekil – 4’de ve kodun çıktısı da Şekil – 4.1’de gösterilmiştir.

```
In [42]: #Checkbutton Kullanımı
import tkinter as tk

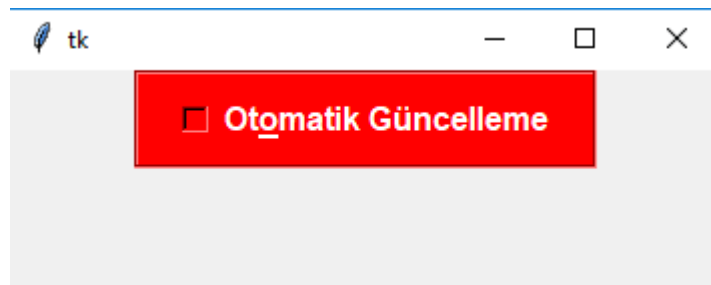
mainFrame = tk.Tk()

def messageCallBack() :
    messagebox.showinfo("Dikkat!", "Bu bir Checkbutton")

mycheckbutton = tk.Checkbutton(mainFrame,
    activebackground = "white",
    activeforeground = "red",
    bg = "red",
    bitmap = "",
    bd = 2,
    command = messageCallBack,
    cursor = "mouse",
    disabledforeground = "purple",
    font = "Arial 12 bold",
    fg = "white",
    height = 2,
    width = 20,
    highlightcolor = "green",
    image = "",
    justify = "center",
    offvalue = 0,
    onvalue = 1,
    padx = 1,
    pady = 1,
    relief = "groove",
    selectcolor = "red",
    selectimage = "",
    state = "normal",
    text = "Otomatik Güncelleme",
    underline = 2,
    wraplength = 230)

mycheckbutton.pack()
mainFrame.mainloop()
```

Şekil – 4



Şekil – 4.1

➤ **Entry**

Kullanıcıdan değerleri kabul etmek için tek satırlık bir metin alanını görüntülemek için kullanılır.

Sözdizimi (Syntax)

```
w = Entry( master, option=value, ... )
```

Parametreler

▪ **master** : Ana pencereyi temsil eder  
**options** : Seçenekler listesini temsil eder. Bu seçenekler virgülle ayrılmış anahtar – değer çiftleri olarak kullanılır. En çok kullanılan seçeneklerin listesi aşağıda belirtilmiştir.

Sr.No.	Seçenek & Açıklama
1	<b>bg</b> Etiket ve göstergenin arkasında görüntülenen normal arka plan rengi.
2	<b>bd</b> Gösterge etrafındaki sınırın boyutu. Varsayılan 2 pikseldir.
3	<b>command</b> Kullanıcının bu kontrol düğmesinin durumunu her değiştirdiğinde çağrılacak fonksiyon adı.
4	<b>cursor</b> Bu seçeneği bir imleç ismine (arrow, dot vb.) ayarlarsanız, fare imleci, kontrol tuşunun üzerindeyken o desene değişir.
5	<b>font</b> Metin için kullanılan yazı tipi.
6	<b>exportselection</b> Varsayılan olarak, bir Giriş widget'ında metin seçerseniz, otomatik olarak panoya aktarılır. Bu aktarımı önlemek için exportselection = 0 kullanın.

7	<b>fg</b> Metni oluşturmak için kullanılan renk.
8	<b>highlightcolor</b> Fokus olunduğunda ki rengi.
9	<b>justify</b> Text birden çok satır içerirse, text'in nereye yanaşık olacağını belirler. CENTER değeri ön değerdir. LEFT ve RIGHT değerleri de verilebilir.
10	<b>relief</b> Varsayılan değer olan relief = FLAT ile, widget arka planında göze çarpmaz. Bu seçeneği diğer stillerden birine ayarlayabilirsiniz.
11	<b>selectbackground</b> Seçilen metni görüntülemek için kullanılacak arka plan rengi.
12	<b>selectborderwidth</b> Seçilen metnin etrafındaki sınırın genişliği. Varsayılan bir pikseldir.
13	<b>selectforeground</b> Seçilen metnin önalın (metin) rengi.
14	<b>show</b> Normalde, kullanıcı türlerinin girişte görüntülediği karakterler. Her karakteri yıldız olarak ekleyen bir şifre girişı yapmak için show = "*" değerini ayarlayın.
15	<b>state</b> Varsayılan durum = NORMAL durumundadır, ancak kontrolü bozmak ve yanıt vermemesi için state = DISABLED ögesini kullanabilirsiniz. İmleç şu anda kontrol düğmesinin üzerindeyse, durum AKTİF.
16	<b>textvariable</b>



	Geçerli metni giriş widget'ınızdan alabilmek için, bu seçeneği StringVar sınıfının bir örneğine ayarlamanız gerekir.
17	<p><b>width</b></p> <p>Bir kontrol düğmesinin varsayılan genişliği, görüntülenen resmin veya metnin boyutuna göre belirlenir. Bu seçeneği bir dizi karaktere ayarlayabilir ve kontrol panelinde her zaman bu kadar çok karakter için yer olacaktır.</p>
18	<p><b>xscrollcommand</b></p> <p>Kullanıcıların genellikle widget'ın ekran boyutundan daha fazla metin girmesini beklerseniz giriş widget'ınızı bir kaydırma çubuğuna bağlayabilirsiniz.</p>

### Metodlar

Checkbutton için en yaygın kullanılan metodlar şunlardır;

Sr.No.	Metod & Açıklama
1	<p><b>delete ( first, last=None )</b></p> <p>Widget'tan karakterleri siler, baştakinden en sonuncuya kadar olan karakterleri içermez. İkinci argüman ihmal edilirse, ilk önce sadece tek karakter silinir.</p>
2	<p><b>get()</b></p> <p>Girişin geçerli metnini bir string olarak döndürür.</p>
3	<p><b>icursor ( index )</b></p> <p>Ekleme işaretçisini verilen endeksteki karakterden hemen önce ayarla.</p>
4	<p><b>index ( index )</b></p> <p>Girişin içeriğini kaydırır, böylece, belirtilen dizindeki karakter en soldaki görünür karakterdir.</p>
5	<p><b>insert ( index, s )</b></p> <p>Verilen index karakterinden önce s stringini ekler.</p>

6	<p><b>select_adjust ( index )</b></p> <p>Bu yöntem, seçimin belirtilen dizindeki karakteri içerdiğinden emin olmak için kullanılır.</p>
7	<p><b>select_clear()</b></p> <p>Seçimi temizler. Seçim yoksa, hiçbir etkisi yoktur.</p>
8	<p><b>select_from ( index )</b></p> <p>ANCHOR indeks pozisyonunu indeks ile seçilen karaktere ayarlar ve bu karakteri seçer.</p>
9	<p><b>select_present()</b></p> <p>Bir seçim varsa, true değerini döndürür, aksi halde false değerini döndürür.</p>
10	<p><b>select_range ( start, end )</b></p> <p>Seçimi program kontrolü altında ayarlar. Başlangıç endeksinden başlayarak, bitiş dizindeki karakteri dahil etmeyecek şekilde başlayan metni seçer. Başlangıç pozisyonu son konumdan önce olmalıdır.</p>
11	<p><b>select_to ( index )</b></p> <p>Tüm metni ANCHOR konumundan itibaren verilen dizindeki karaktere kadar değil de dahil olmak üzere seçer.</p>
12	<p><b>xview ( index )</b></p> <p>Bu yöntem Entry widget'inin yatay bir kaydırma çubuğuna bağlanması için kullanışlıdır.</p>
13	<p><b>xview_scroll ( number, what )</b></p> <p>Girişi yatay olarak kaydırmak için kullanılır. Hangi argümanın UNITS, karakter genişliklerine göre kaydırılması veya PAGE ile giriş widget'inin büyüklüğüne göre kaydırılması gerekir. Soldan sağa kaydırmak için pozitif, sağdan sola kaydırmak için negatiftir.</p>

Entry kullanımına ait örnek kod Şekil – 5’de ve kodun çıktısı da Şekil – 5.1’de gösterilmiştir.

```
In [37]: #Entry Kullanımı
import tkinter as tk
from tkinter import messagebox

mainFrame = tk.Tk()

def kullanıcıGirisKontrol() :
    messagebox.showinfo("Dikkat!", "Kullanıcı Adı : " + txtKullaniciAdi.get() + "\nŞifre : " + txtSifre.get())

lblKullaniciAdi = tk.Label(mainFrame, text="Kullanıcı Adı : ")
txtKullaniciAdi = tk.Entry(mainFrame, bg = "red", bd = 2, font = "Arial 12 bold", fg = "white")

lblSifre = tk.Label(mainFrame, text="Şifre : ")
txtSifre = tk.Entry(mainFrame, bg = "red", bd = 2, font = "Arial 12 bold", fg = "white", show = "**")

btnGiris = tk.Button(mainFrame, text = "Giriş", command = kullanıcıGirisKontrol)

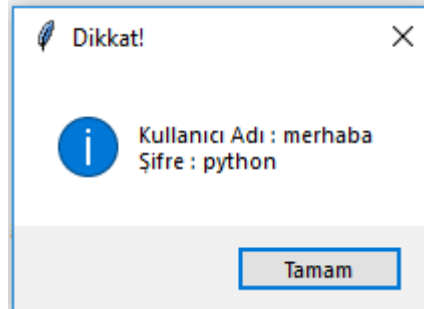
#Herbir nesnenin ekrandaki yerini belirle
lblKullaniciAdi.grid(row=0, column=0, padx=5, pady=5, sticky = "E") #Sağa hizalı (sticky = "E")
txtKullaniciAdi.grid(row=0, column=1, padx=5, pady=5, sticky = "W") #Sola hizalı (sticky = "W")

lblSifre.grid(row=1, column=0, padx=5, pady=5, sticky = "E")
txtSifre.grid(row=1, column=1, padx=5, pady=5, sticky = "W")

btnGiris.grid(row=2, columnspan=2, padx=5, pady=5)

mainFrame.mainloop()
```

Şekil – 5



Şekil - 5.1

➤ **Frame**

Diğer widget'ları düzenlemek için bir konteyner widget'ı olarak kullanılır.

Sözdizimi (Syntax)

```
w = Frame( master, option=value, ... )
```

Parametreler

- **master** : Ana pencereyi temsil eder
- **options** : Seçenekler listesini temsil eder. Bu seçenekler virgülle ayrılmış anahtar – değer çiftleri olarak kullanılır. En çok kullanılan seçeneklerin listesi aşağıda belirtilmiştir.

Sr.No.	Seçenek & Açıklama
1	<b>bg</b> Çerçevenin zemin rengi.
2	<b>bd</b> Çerçevenin sınır çizgilerinin kalınlığı. Bir değer atanmamış ise öndeğeri (default) 2 pixel'dir.
3	<b>Cursor</b> Fare işaretçisinin biçimini seçer. Fareyle imleç çerçevedeki checkbutton widgetı üzerine konulunca, seçilen imleç biçimi görünür.
4	<b>height</b> Çerçevenin yüksekliği.
5	<b>highlightbackground</b> Fare ile üzerine gidildiğinde çerçevenin zemini aydınlanır.
6	<b>highlightcolor</b> Fare ile üzerine gidildiğinde çerçevenin aydınlığı için renk seçilir.
7	<b>highlightthickness</b> Aydınlanmanın yoğunluğunu seçer.

8	<p><b>relief</b></p> <p>FLAT, RAISED, SUNKEN, GROOVE, RIDGE (düz, kabartma, oyma, oluk kenarlı, sırt kenarlı) biçimlerinden birisi seçilebilir. Öndeğer rölyef relief=FLAT olur. Rölyef seçenekleri checkbutton dışındaki bütün widgetlere uygulanabilir. Ancak, checkbutton kendi zemininden yukarıya çıkmaz.</p>
9	<p><b>width</b></p> <p>Çerçevenin genişliğini belirler. Çerçeve, kendi üzerine konulan widgetlere göre kendi genişliğini belirler. Checkbutton genişliği, üzerine konulan ikon yada resimlere göre kendiliğinden ayarlanır.</p>

Frame kullanımına ait örnek kod Şekil – 6’de ve kodun çıktısı da Şekil – 6.1’de gösterilmiştir.

```
# Frame Kullanımı
import tkinter as tk

# Fonksiyonlar
def center_window(width=300, height=200):
    # Ekran genişliği ve yüksekliği
    screen_width = win.winfo_screenwidth()
    screen_height = win.winfo_screenheight()

    # x ve y koordinatlarının değerini hesapla
    x = (screen_width / 2) - (width / 2)
    y = (screen_height / 2) - (height / 2)
    win.geometry('%dx%d+%d+%d' % (width, height, x, y))

win = tk.Tk()
win.title("Frame Uygulaması")
center_window(300, 300)

frame = tk.Frame(win, bg="yellow", width=200, height=200)
frame.pack(fill=tk.BOTH, expand=1)

bottomframe = tk.Frame(win, bg="red", width=200, height=200)
bottomframe.pack(fill=tk.BOTH, expand=1)

redbutton = tk.Button(frame, text="Red", fg="red", padx=5, pady=5)
redbutton.grid(row=0, column=0)

greenbutton = tk.Button(frame, text="Brown", fg="brown", padx=5, pady=5)
greenbutton.grid(row=0, column=1)

bluebutton = tk.Button(frame, text="Blue", fg="blue", padx=5, pady=5)
bluebutton.grid(row=0, column=2)

blackbutton = tk.Button(bottomframe, text="Black", fg="black", padx=5, pady=5)
blackbutton.grid(row=0, column=0)

win.mainloop()
```

Şekil – 6



Şekil – 6.1

➤ **Label**

Diğer widget'lar için tek satırlık yazı sağlamak için kullanılır. Ayrıca görüntüler de içerebilir.

Sözdizimi (Syntax)

```
w = Label( master, option=value, ... )
```

Parametreler

- **master** : Ana pencereyi temsil eder
- **options** : Seçenekler listesini temsil eder. Bu seçenekler virgülle ayrılmış anahtar – değer çiftleri olarak kullanılır. En çok kullanılan seçeneklerin listesi aşağıda belirtilmiştir.

Sr.No.	Seçenek & Açıklama
1	<b>anchor</b> Etiket, metin ihtiyacından daha fazla alana sahipse, bu seçenekler metnin nerede konumlandırıldığını kontrol eder. Varsayılan, metnin kullanılabilir alanda ortalanmasını sağlayan anchor = CENTER'dir.
2	<b>bg</b> Etiket'in zemin rengi.
3	<b>bitmap</b> Bu seçeneği bir bitmap veya resim nesnesine eşit olarak ayarlayın ve etiket bu grafiği gösterecektir.
4	<b>bd</b> Etiket'in sınır çizgilerinin kalınlığı. Bir değer atanmamış ise öndeğeri (default) 2 pixel'dir.
5	<b>cursor</b> Bu seçeneği bir imleç ismine (arrow, dot, vb.) ayarlarsanız, fare imleci, kontrol tuşunun üzerindeyken o desene değişir.
6	<b>font</b>

	Metin için kullanılan yazı tipi.
7	<p><b>fg</b></p> <p>Bu etikette metin veya bitmap görüntülüyorsanız, bu seçenek metnin rengini belirtir. Bir bitmap görüntülüyorsanız, bu bitmap'teki 1 bitlerin konumunda görünecek olan renktir.</p>
8	<p><b>height</b></p> <p>Etiketin yüksekliği.</p>
9	<p><b>image</b></p> <p>Etiket widget'ında statik bir görüntüyü görüntülemek için, bu seçeneği bir görüntü nesnesine ayarlayın.</p>
10	<p><b>justify</b></p> <p>Text birden çok satır içerirse, text'in nereye yanaşık olacağını belirler. CENTER değeri ön değerdir. LEFT ve RIGHT değerleri de verilebilir.</p>
11	<p><b>padx</b></p> <p>Etiket içerisindeki metnin soluna ve sağına fazladan boşluk ekler. Varsayılan 1'dir.</p>
12	<p><b>pady</b></p> <p>Etiket içerisindeki metnin üstünde ve altında ekstra boşluk ekler. Varsayılan 1'dir.</p>
13	<p><b>relief</b></p> <p>Etiketin etrafında dekoratif bir sınırın görünümünü belirtir. Varsayılan FLAT'tır.</p>
14	<p><b>text</b></p> <p>Bir etiket widget'ındaki bir veya daha fazla metin satırını görüntülemek için, bu seçeneği metni içeren bir dizeye ayarlayın. Yeni satırlar ("\\n") için kullanılır.</p>

15	<p><b>textvariable</b></p> <p>Bir etiket widget'ında görüntülenen metni StringVar sınıfının bir kontrol değişkenine bağlamak için bu seçeneği o değişkene ayarlayın.</p>
16	<p><b>underline</b></p> <p>Bu seçeneği n olarak ayarlayarak metnin nth harfinin altında bir alt çizgi ( _ ) görüntüleyerek 0'dan sayabilirsiniz. Varsayılan alt çizgi = -1'dir, yani alt çizgi yoktur.</p>
17	<p><b>width</b></p> <p>Etiket içindeki karakterlerin genişliği (piksel değil!). Bu seçenek belirlenmemişse, etiket içeriğine uyacak şekilde boyutlandırılacaktır.</p>
18	<p><b>wraplength</b></p> <p>Bu seçeneği istenen sayıya ayarlayarak her satırdaki karakter sayısını sınırlandırabilirsiniz. Varsayılan değer, 0, satırların yalnızca yeni satırlarda kırılacağı anlamına gelir.</p>

Label kullanımına ait örnek kod Şekil – 7’de ve kodun çıktısı da Şekil – 7.1’de gösterilmiştir.

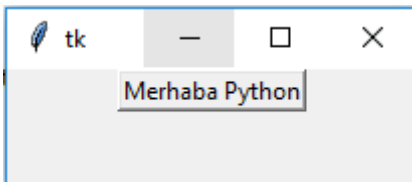
```

In [54]: #Label Kullanımı
import tkinter as tk

root = tk.Tk()
label = tk.Label( root, text="Merhaba Python", relief="raised" )
label.pack()
root.mainloop()

```

Şekil – 7



Şekil – 7.1



### ➤ Listbox

Kullanıcıya seçeneklerin listesini sağlamak için kullanılır.

#### Sözdizimi (Syntax)

```
w = Listbox( master, option=value, ... )
```

#### Parametreler

- **master** : Ana pencereyi temsil eder
- **options** : Seçenekler listesini temsil eder. Bu seçenekler virgülle ayrılmış anahtar – değer çiftleri olarak kullanılır. En çok kullanılan seçeneklerin listesi aşağıda belirtilmiştir.

Sr.No.	Seçenek & Açıklama
1	<b>bg</b> Etiket ve göstergenin arkasında görüntülenen normal arka plan rengi.
2	<b>bd</b> Gösterge etrafındaki sınırın boyutu. Varsayılan 2 pikseldir.
3	<b>cursor</b> Fare, liste kutusunun üzerinde olduğunda görüntülenen imleç.
4	<b>font</b> Liste kutusundaki metin için kullanılan yazı tipi.
5	<b>fg</b> Liste kutusundaki metin için kullanılan renk.
6	<b>height</b> Liste kutusunda gösterilen satır sayısı (piksel değil!). Varsayılan 10'dur.
7	<b>highlightcolor</b> Widget odakta olduğunda odakta gösterilen renk vurgulanır.
8	<b>highlightthickness</b>

	Odak vurgulama kalınlığı.
9	<p><b>relief</b></p> <p>Üç boyutlu kenar gölgelendirme efektlerini seçer. Varsayılan SUNKEN'dir.</p>
10	<p><b>selectbackground</b></p> <p>Seçilen metni görüntülemek için kullanılacak arka plan rengi.</p>
11	<p><b>selectmode</b></p> <p>Kaç öğenin seçilebileceğini ve fare sürüklemelerinin seçimi nasıl etkilediğini belirler –</p> <ul style="list-style-type: none"> <li>• <b>BROWSE</b> – Normal olarak, bir liste kutusundan sadece bir satır seçebilirsiniz. Bir öğeye tıklar ve farklı bir satıra sürüklerseniz, seçim fareyi izler. Bu varsayılandır.</li> <li>• <b>SINGLE</b> – Sadece bir satır seçebilir ve mouse'u sürükleyemezsiniz. 1 butona bastığınızda, o satır seçilir.</li> <li>• <b>MULTIPLE</b> – Bir kerede herhangi bir sayıda satır seçebilirsiniz. Herhangi bir satıra tıklamak, seçilip seçilmediğini değiştirir.</li> <li>• <b>EXTENDED</b> – İlk satıra tıklayarak ve son satıra sürükleyerek herhangi bir bitişik satır grubunu bir kerede seçebilirsiniz.</li> </ul>
12	<p><b>width</b></p> <p>Widget'ın karakterlerde genişliği. Varsayılan 20'dir.</p>
13	<p><b>xscrollcommand</b></p> <p>Kullanıcının liste kutusunu yatay olarak kaydırmasına izin vermek istiyorsanız, liste kutusu widget'ınızı yatay kaydırma çubuğuna bağlayabilirsiniz.</p>
14	<p><b>yscrollcommand</b></p> <p>Kullanıcının liste kutusunu dikey olarak kaydırmasına izin vermek istiyorsanız, liste kutusu widget'ınızı dikey kaydırma çubuğuna bağlayabilirsiniz.</p>

### Metodlar

Listbox için en yaygın kullanılan metodlar şunlardır;

Sr.No.	Metod & Açıklama
1	<b>activate ( index )</b> Belirtilen index'e göre satırları seçer.
2	<b>curselection()</b> Seçilen elemanın veya elemanların satır numaralarını içeren, 0'dan sayılan bir tuple döndürür. Hiçbir şey seçilmezse boş bir tuple döndürür.
3	<b>delete ( first, last=None )</b> Endeksleri [ilk, son] aralığında olan satırları siler. İkinci argüman ihmal edilirse, önce indeksi olan tek satır silinir.
4	<b>get ( first, last=None )</b> İlk satırdan sonuna kadar olan satırları içeren satırların metnini içeren bir tuple döndürür. İkinci bağımsız değişken atlanırsa, en yakın satırın metnini ilk durumuna döndürür.
5	<b>index ( i )</b> Mümkünse, liste kutusunun görünür kısmını, indeks i içeren satırın widget'ın üstünde olmasını sağlar.
6	<b>insert ( index, *elements )</b> Index ile belirtilen satırın önüne bir veya daha fazla yeni satırı liste kutusuna ekler. Liste kutusunun sonuna yeni satırlar eklemek istiyorsanız, ilk bağımsız değişken olarak END kullanın.
7	<b>nearest ( y )</b> Liste kutusu widget'ına göre y koordinatına en yakın görülebilir satırın index'ini döndürür.

8	<p><b>see ( index )</b></p> <p>Liste kutusunun konumunu ayarlayın, böylece endekse atıfta bulunulan satır görünür.</p>
9	<p><b>size()</b></p> <p>Liste kutusundaki satır sayısını döndürür.</p>
10	<p><b>xview()</b></p> <p>Liste kutusunu yatay kaydırılabilir yapmak için, ilişkili yatay kaydırma çubuğunun komut seçeneğini bu yöntemle ayarlar.</p>
11	<p><b>xview_moveto ( fraction )</b></p> <p>Scroll the listbox so that the leftmost fraction of the width of its longest line is outside the left side of the listbox. Fraction is in the range [0,1].</p>
12	<p><b>xview_scroll ( number, what )</b></p> <p>Liste kutusunu yatay olarak kaydırır. What argümanı için, karakterlere göre kaydırmak için UNITS veya sayfalara göre kaydırmak için PAGES kullanın, yani liste kutusunun genişliği ile. Number argümanı kaç tane kaydırılacağını söyler.</p>
13	<p><b>yview()</b></p> <p>Liste kutusunu dikey olarak kaydırılabilir yapmak için, ilişkili dikey kaydırma çubuğunun komut seçeneğini bu yöntemle ayarlayın.</p>
14	<p><b>yview_moveto ( fraction )</b></p> <p>Scroll the listbox so that the top fraction of the width of its longest line is outside the left side of the listbox. Fraction is in the range [0,1].</p>
15	<p><b>yview_scroll ( number, what )</b></p> <p>Liste kutusunu dikey olarak kaydırır. What argümanı için, karakterlere göre kaydırmak için UNITS veya sayfalara göre kaydırmak için PAGES kullanın, yani liste kutusunun genişliği ile. Number argümanı kaç tane kaydırılacağını söyler.</p>

Listbox kullanımına ait örnek kod Şekil – 8’de ve kodun çıktısı da Şekil – 8.1’de gösterilmiştir.

```
In [40]: import tkinter as tk

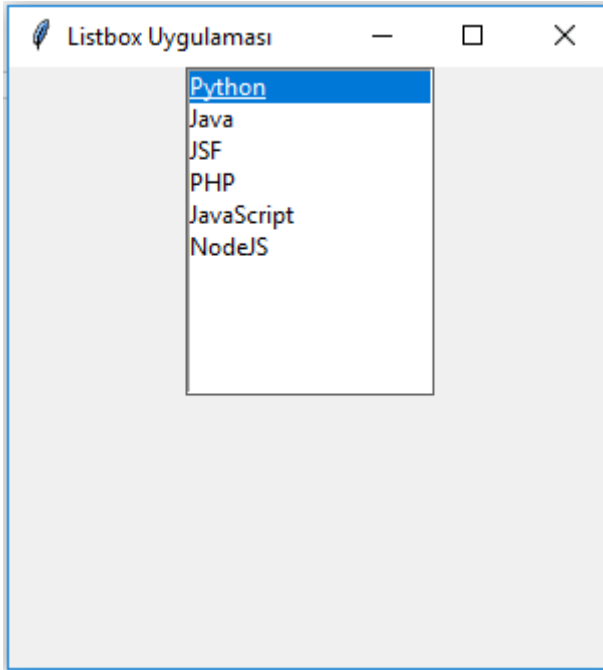
win = tk.Tk()
win.title("Listbox Uygulaması")
center_window(300, 300)

#Listbox nesnesini çiz
Lb1 = tk.Listbox(win)
Lb1.insert(1, "Python")
Lb1.insert(2, "Java")
Lb1.insert(3, "JSF")
Lb1.insert(4, "PHP")
Lb1.insert(5, "JavaScript")
Lb1.insert(6, "NodeJS")

Lb1.pack()

#Formu çiz
win.mainloop()
```

Şekil – 8



Şekil – 8.1

➤ **Menubutton**

Uygulamanızdaki menüleri görüntülemek için kullanılır.

Sözdizimi (Syntax)

```
w = Menubutton( master, option=value, ... )
```

Parametreler

- **master** : Ana pencereyi temsil eder
- **options** : Seçenekler listesini temsil eder. Bu seçenekler virgülle ayrılmış anahtar – değer çiftleri olarak kullanılır. En çok kullanılan seçeneklerin listesi aşağıda belirtilmiştir.

Sr.No.	Seçenek & Açıklama
1	<b>activebackground</b> Fare menubuttonun üzerindeyken arka plan rengi.
2	<b>activeforeground</b> Fare menubuttonun üzerindeyken ön plan rengi.
3	<b>anchor</b> Bu seçenek kontrolleri, pencere ögesi metin gereksiniminden daha fazla alana sahipse metnin nerede konumlandırılacağını kontrol eder. Varsayılan, metni merkezleyen anchor = CENTER'dir.
4	<b>bg</b> Etiket ve göstergenin arkasında görüntülenen normal arka plan rengi.
5	<b>bitmap</b> Menubuttonda bir bitmap görüntülemek için bu seçeneği bir bitmap adına ayarlayın.
6	<b>bd</b> Gösterge etrafındaki sınırın boyutu. Varsayılan 2 pikseldir.
7	<b>cursor</b>

	Fare bu menubuttonun üzerine geldiğinde görünen imleç.
8	<p><b>direction</b></p> <p>Menüyü, düğmenin soluna görüntülemek için direction = LEFT ayarlayın; Menüye sağ tuşa basarak görüntülemek için direction = RIGHT; ya da düğmenin üstündeki menüyü yerleştirmek direction = ABOVE yönünü kullanın.</p>
9	<p><b>disabledforeground</b></p> <p>Devre dışı bırakıldığında bu menüde gösterilen ön plan rengi.</p>
10	<p><b>fg</b></p> <p>Fare menubuttonun üzerinde değilken ön plan rengi.</p>
11	<p><b>height</b></p> <p>Text'in satırlarındaki menubutton'un yüksekliği (piksel değil!). Varsayılan, menubutton'un boyutunu içeriğine sığdırmaktır.</p>
12	<p><b>highlightcolor</b></p> <p>Widget odakta olduğunda odakta gösterilen renk vurgulanır.</p>
13	<p><b>image</b></p> <p>Bu menünün üzerindeki bir image'ı görüntülemek için.</p>
14	<p><b>justify</b></p> <p>Bu seçenek, metin menubuttonu doldurmadığında metnin nerede bulunduğunu kontrol eder. Metni sola hizalamak için justify = LEFT (bu varsayılan değerdir); ortalamak için justify = CENTER, sağ hizalamak için justify = RIGHT kullanılır.</p>
15	<p><b>menu</b></p> <p>Menüyü bir dizi seçenekle ilişkilendirmek için bu seçeneği, bu seçenekleri içeren Menü nesnesine ayarlayın. Bu menü nesnesi, ilişkili menüyü ilk argüman olarak kurucuya geçirerek oluşturulmuş olmalıdır.</p>

16	<p><b>padx</b></p> <p>Menubutton metninin soluna ve sağına ne kadar boşluk bırakılır. Varsayılan 1'dir.</p>
17	<p><b>pady</b></p> <p>Menubuttonun metninin üstünde ve altında ne kadar boşluk bırakılır. Varsayılan 1'dir.</p>
18	<p><b>relief</b></p> <p>Üç boyutlu kenar gölgelendirme efektlerini seçer. Varsayılan RAISED.</p>
19	<p><b>state</b></p> <p>Normalde menubuttons, fareye cevap verir. Menubutton gri hale getirmek ve yanıt vermemek için state=DISABLED ayarlayın.</p>
20	<p><b>text</b></p> <p>Menubutton üzerindeki metni görüntülemek için, bu seçeneği istenen metni içeren dizeye ayarlayın. Dizenin içindeki yeni satırlar (" \ n") satır sonlarına neden olur.</p>
21	<p><b>textvariable</b></p> <p>StringVar sınıfının bir kontrol değişkenini bu menubutton ile ilişkilendirebilirsiniz. Bu kontrol değişkenini ayarlamak, görüntülenen metni değiştirecektir.</p>
22	<p><b>underline</b></p> <p>Normalde menubutton metnin altında alt çizgi görünmez. Karakterlerden birinin altını çizmek için, bu seçeneği o karakterin dizinine ayarlayın.</p>
23	<p><b>width</b></p> <p>Widget'ın karakterlerde genişliği. Varsayılan 20'dir.</p>
24	<p><b>wrplength</b></p>



Normalde, satırlar kaydırılmaz. Bu seçeneği bir dizi karaktere ayarlayabilir ve tüm satırlar bu sayıdan daha uzun olmayan parçalara ayrılacaktır.

Menubutton kullanımına ait örnek kod Şekil – 9’de ve kodun çıktısı da Şekil – 9.1’de gösterilmiştir.

```
In [11]: import tkinter as tk

top = tk.Tk()

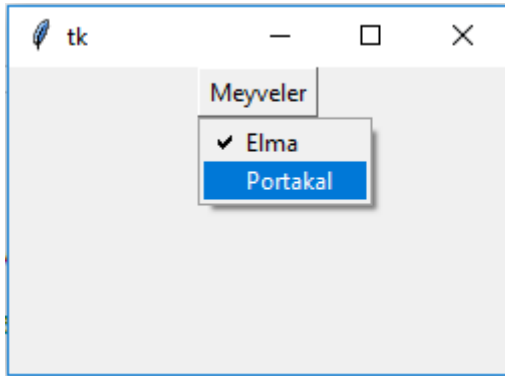
mb = tk.Menubutton ( top, text="Meyveler", relief="raised" )
mb.grid()
mb.menu = tk.Menu ( mb, tearoff = 0 )
mb["menu"] = mb.menu

mayoVar = tk.IntVar()
ketchVar = tk.IntVar()

mb.menu.add_checkbutton ( label="Elma", variable=mayoVar )
mb.menu.add_checkbutton ( label="Portakal", variable=ketchVar )

mb.pack()
top.mainloop()
```

Şekil – 9



Şekil – 9.1

➤ **Menu**

Kullanıcıya çeşitli komutlar sağlamak için kullanılır. Bu komutlar Menubutton içinde yer almaktadır.

Sözdizimi (Syntax)

```
w = Menu ( master, option=value, ... )
```

Parametreler

- **master** : Ana pencereyi temsil eder
- **options** : Seçenekler listesini temsil eder. Bu seçenekler virgülle ayrılmış anahtar – değer çiftleri olarak kullanılır. En çok kullanılan seçeneklerin listesi aşağıda belirtilmiştir.

Sr.No.	Seçenek & Açıklama
1	<b>activebackground</b> Farenin altında bir seçim üzerinde görünecek arka plan rengi.
2	<b>activeborderwidth</b> Farenin altında bir seçim etrafında çizilmiş bir kenarlık genişliğini belirtir. Varsayılan 1 pikseldir.
3	<b>activeforeground</b> Farenin altında bir seçenek üzerinde görünecek ön plan rengi.
4	<b>bg</b> Farenin altında olmayan seçenekler için arka plan rengi.
5	<b>bd</b> Tüm seçeneklerin etrafındaki sınırın genişliği. Varsayılan 1'dir.
6	<b>cursor</b> Fare, seçenekler bittiğinde görünen ancak yalnızca menü kapatıldığında görünen imleç.
7	<b>disabledforeground</b>

	Durumu DISABLED olan öğeler için metnin rengi.
8	<b>font</b> Metin seçimleri için varsayılan yazı tipi.
9	<b>fg</b> Ön plan rengi, farenin altında olmayan seçenekler için kullanılır.
10	<b>postcommand</b> Bu seçeneği bir prosedüre ayarlayabilirsiniz ve bu menüde bir kişi her seferinde bu prosedürü çağırır.
11	<b>relief</b> Menüler için varsayılan 3-D efekti relief = RAISED.
12	<b>image</b> Bu menubutton üzerindeki bir görüntüyü görüntülemek için.
13	<b>selectcolor</b> Seçilen checkbutton ve radiobutton'da görüntülenen rengi belirtir.
14	<b>tearoff</b> Normalde, bir menü kapatılabilir, seçenekler listesindeki ilk konum (konum 0) koparma öğesi tarafından kaplanır ve ek seçimler konum 1'den başlayarak eklenir. Eğer tearoff = 0'ı ayarlarsanız, menü koparma özelliğine sahip olmayacak ve seçenekler 0'dan itibaren eklenecektir.
15	<b>title</b> Normalde, bir yırtma menü penceresinin başlığı, menu veya menubutton kaskadı metniyle aynı olacaktır. Bu pencerenin başlığını değiştirmek isterseniz, başlık dizisini bu dizeye ayarlayın.

### Metodlar

Menu için en yaygın kullanılan metodlar şunlardır;

Sr.No.	Metod & Açıklama
1	<b>add_command (options)</b> Menüye bir menü öğesi ekler.
2	<b>add_radiobutton( options )</b> Bir radyo düğmesi menü öğesi oluşturur.
3	<b>add_checkbutton( options )</b> Bir kontrol düğmesi menü öğesi oluşturur.
4	<b>add_cascade(options)</b> Belirli bir menüyü üst menüye ilişkilendirerek yeni bir hiyerarşik menü oluşturur.
5	<b>add_separator()</b> Menüye bir ayırıcı çizgi ekler.
6	<b>add( type, options )</b> Menüye belirli bir menü öğesi türü ekler.
7	<b>delete( startindex [, endindex ])</b> startindex ile endindex arasında değişen menü öğelerini siler.
8	<b>entryconfig( index, options )</b> Index ile tanımlanan bir menü öğesini değiştirmenize ve seçeneklerini değiştirmenize izin verir.
9	<b>index(item)</b> Verilen menü öğesi etiketinin indeks numarasını döndürür.

10	<b>insert_separator ( index )</b>  İndeks ile belirtilen konuma yeni bir ayırıcı yerleştirin.
11	<b>invoke ( index )</b>  İndex konumundaki seçim ile ilişkili komut geri çağırır. Bir checkbox ise durumu set ve cleared arasında değişir; Bir radiobutton ise bu seçenek belirlenir.
12	<b>type ( index )</b>  İndex ile belirtilen seçimin türünü döndürür: "cascade", "checkboxbutton", "command", "radiobutton", "seperator" veya "tearoff".

Menu kullanımına ait örnek kod Şekil – 10’da ve kodun çıktısı da Şekil – 10.1’de gösterilmiştir.

```
import tkinter as tk

def donothing():
    filewin = tk.Toplevel(root)
    button = tk.Button(filewin, text="Do nothing button")
    button.pack()

root = tk.Tk()
menubar = tk.Menu(root)
filemenu = tk.Menu(menubar, tearoff=0)
filemenu.add_command(label="New", command=donothing)
filemenu.add_command(label="Open", command=donothing)
filemenu.add_command(label="Save", command=donothing)
filemenu.add_command(label="Save as...", command=donothing)
filemenu.add_command(label="Close", command=donothing)

filemenu.add_separator()

filemenu.add_command(label="Exit", command=root.quit)
menubar.add_cascade(label="File", menu=filemenu)
editmenu = tk.Menu(menubar, tearoff=0)
editmenu.add_command(label="Undo", command=donothing)

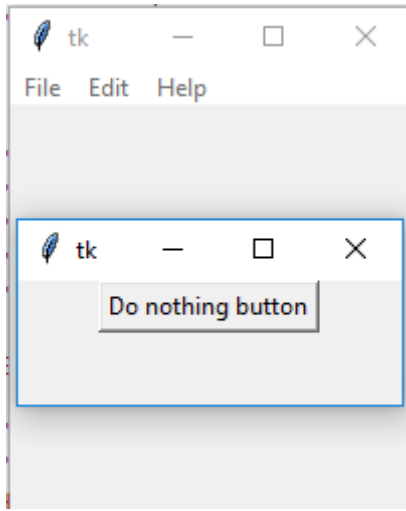
editmenu.add_separator()

editmenu.add_command(label="Cut", command=donothing)
editmenu.add_command(label="Copy", command=donothing)
editmenu.add_command(label="Paste", command=donothing)
editmenu.add_command(label="Delete", command=donothing)
editmenu.add_command(label="Select All", command=donothing)

menubar.add_cascade(label="Edit", menu=editmenu)
helpmenu = tk.Menu(menubar, tearoff=0)
helpmenu.add_command(label="Help Index", command=donothing)
helpmenu.add_command(label="About...", command=donothing)
menubar.add_cascade(label="Help", menu=helpmenu)

root.config(menu=menubar)
root.mainloop()
```

Şekil – 10



Şekil – 10.1

#### ➤ Message

Kullanıcıdan değerleri kabul etmek için çok satırlı metin alanlarını görüntülemek için kullanılır.

#### Sözdizimi (Syntax)

```
w = Message ( master, option=value, ... )
```

#### Parametreler

- **master** : Ana pencereyi temsil eder
- **options** : Seçenekler listesini temsil eder. Bu seçenekler virgülle ayrılmış anahtar – değer çiftleri olarak kullanılır. En çok kullanılan seçeneklerin listesi aşağıda belirtilmiştir.

Sr.No.	Seçenek & Açıklama
1	<b>anchor</b> Eğer widget, metin ihtiyacından daha fazla alana sahipse, bu seçenek metnin nerede konumlandırıldığını kontrol eder. Varsayılan, metnin kullanılabilir alanda ortalanmasını sağlayan anchor = CENTER'dir.
2	<b>bg</b> Etiket ve göstergenin arkasında görüntülenen normal arka plan rengi.
3	<b>bitmap</b> Bu seçeneği bir bitmap veya resim nesnesine eşit olarak ayarlayın ve etiket bu

	grafięi g�sterecektir.
4	<b>bd</b> G�sterge etrafındaki sınırın boyutu. Varsayılan 2 pikseldir.
5	<b>cursor</b> Bu seeneęi bir imle ismine (arrow, dot, vb.) ayarlarsanız, fare imleci, kontrol tuşunun �zerindeyken o desene deęiřir.
6	<b>font</b> text veya textvariable seeneęi ile bu etikette metin g�r�nt�l�yorsanız, yazı tipi seeneęi metnin hangi yazı tipinde g�r�nt�leneceęini belirtir.
7	<b>fg</b> Bu etikette metin veya bitmap g�r�nt�l�yorsanız, bu seenek metnin rengini belirtir. Bir bitmap g�r�nt�l�yorsanız, bu bitmap'teki 1 bitlerin konumunda g�r�necek olan renktir.
8	<b>justify</b> Birden fazla metin satırının birbirine g�re nasıl hizalanacaęını belirtir: Sıfır sol iin LEFT, ortalanmıř iin CENTER (varsayılan) veya saęa yaslanmış iin RIGHT.
9	<b>padx</b> Widget ierisindeki metnin soluna ve saęına fazladan bořluk eklenir. Varsayılan 1'dir.
10	<b>pady</b> Widget ierisindeki metnin �st�nde ve altında ekstra bořluk eklenir. Varsayılan 1'dir.
11	<b>relief</b> Etiketin etrafında dekoratif bir sınırın g�r�n�m�n� belirtir. Varsayılan, FLAT.

12	<p><b>text</b></p> <p>Bir etiket widget'ındaki bir veya daha fazla metin satırını görüntülemek için, bu seçeneği metni içeren bir string'e ayarlayın. Bir alt satıra geçmek için ("<code>\ n</code>") kullanılır.</p>
13	<p><b>textvariable</b></p> <p>Bir etiket widget'ında görüntülenen metni StringVar sınıfının bir kontrol değişkenine bağlamak için bu seçeneği o değişkene ayarlayın.</p>
14	<p><b>width</b></p> <p>Etiket içindeki karakterlerin genişliği (piksel değil!). Bu seçenek belirlenmemişse, etiket içeriğine uyacak şekilde boyutlandırılacaktır.</p>

Menu kullanımına ait örnek kod Şekil – 11'da ve kodun çıktısı da Şekil – 11.1'de gösterilmiştir.

```

In [8]: #Message Kullanım
import tkinter as tk

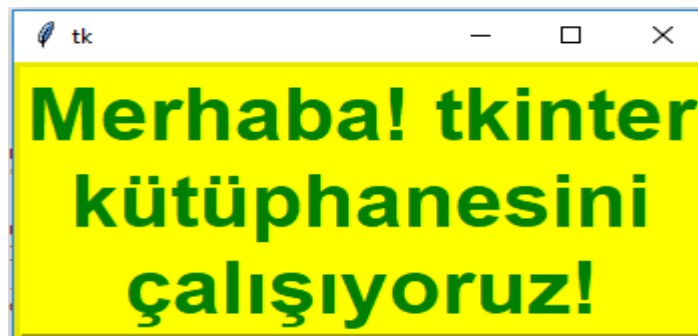
win = tk.Tk()
var = tk.StringVar()
label = tk.Message( win,
                    anchor = "center",
                    bg = "yellow",
                    bd = 5,
                    cursor = "mouse",
                    font = "Arial 32 bold",
                    fg = "green",
                    justify = "center",
                    padx = 1,
                    pady = 1,
                    relief = "raised",
                    text = "textvariable değeri bu değeri ezer",
                    textvariable = var,
                    width = 500 )

var.set("Merhaba! tkinter kütüphanesini çalışıyoruz!")

label.pack()
win.mainloop()

```

Şekil – 11



Şekil – 11.1



➤ **Radiobutton**

Radyo düğmeleri olarak bir dizi seçeneği görüntülemek için kullanılır. Kullanıcı bir seferde sadece bir seçenek seçebilir.

Sözdizimi (Syntax)

```
w = Radiobutton ( master, option=value, ... )
```

Parametreler

- **master** : Ana pencereyi temsil eder
- **options** : Seçenekler listesini temsil eder. Bu seçenekler virgülle ayrılmış anahtar – değer çiftleri olarak kullanılır. En çok kullanılan seçeneklerin listesi aşağıda belirtilmiştir.

Sr.No.	Seçenek & Açıklama
1	<b>activebackground</b> Fare radiobutton nesnesinin üzerindeyken arka plan rengi.
2	<b>activeforeground</b> Fare radiobutton nesnesinin üzerindeyken ön plan rengi.
3	<b>anchor</b> Widget, ihtiyaç duyduğundan daha büyük bir alanda yaşarsa, bu seçenek, radiobutton nesnesinin o alana nerede oturacağını belirtir. Varsayılan anchor = CENTER.
4	<b>bg</b> Göstergenin ve etiketin arkasındaki normal arka plan rengi.
5	<b>bitmap</b> Bir radiobutton nesnesinin üzerindeki tek renkli bir resmi görüntülemek için, bu seçeneği bir bitmap'e ayarlayın.
6	<b>borderwidth</b> Gösterge parçasının etrafındaki kenarlığın boyutu. Varsayılan 2 pikseldir.

7	<b>command</b> Kullanıcı bu radiobutton nesnesinin durumunu her değiştirdiğinde çağrılacak bir prosedür.
8	<b>cursor</b> Bu seçeneği bir imleç ismine (arrow, dot, vb.) ayarlarsanız, fare imleci radiobutton nesnesinin üzerindeyken o desene değişir.
9	<b>font</b> Metin için kullanılan yazı tipi.
10	<b>fg</b> Metni oluşturmak için kullanılan renk.
11	<b>height</b> radiobutton nesnesindeki metinlerin(piksel değil) sayısı. Varsayılan 1'dir.
12	<b>highlightbackground</b> Odaklama rengi, radiobutton nesnesi odaklanmadığında vurgulanır.
13	<b>highlightcolor</b> Odaklanmanın rengi, radiobutton nesnesine odaklanıldığı zaman vurgulanır.
14	<b>image</b> Bu radiobutton nesnesi için metin yerine bir grafik görüntüsü görüntülemek için, bu seçeneği bir image nesnesine ayarlayın.
15	<b>justify</b> Metin birden çok satır içeriyorsa, bu seçenek metnin nasıl hizalanacağını belirler: CENTER (varsayılan), LEFT veya RIGHT.
16	<b>padx</b> Radiobutton nesnesinin ve metnin soluna ve sağına ne kadar boşluk

	birakılacağını belirler. Varsayılan 1'dir.
17	<p><b>pady</b></p> <p>Radiobutton nesnesinin ve metnin üstünde ve altında ne kadar boşluk bırakılacağını belirler. Varsayılan 1'dir.</p>
18	<p><b>relief</b></p> <p>Etiket etrafında dekoratif bir sınırın görünümünü belirtir. Varsayılan, FLAT'tır.</p>
19	<p><b>selectcolor</b></p> <p>Radiobutton nesnesinin set edildiği zamanki rengi. Varsayılan kırmızıdır.</p>
20	<p><b>Selectimage</b></p> <p>Radiobutton nesnesinde metin yerine bir image gösteriliyorsa, Radiobutton nesnesi seçildiğinde görüntülenecek olan image'ı belirtir.</p>
21	<p><b>state</b></p> <p>Radiobutton nesnesini aktif yapmak için state=NORMAL (varsayılan), pasif yapmak için state=DISABLED değerleri kullanılır.</p>
22	<p><b>Text</b></p> <p>Radiobutton nesnesinin yanında görüntülenen etiket. Birden fazla metin satırı görüntülemek için (" \ n") kullanılır.</p>
23	<p><b>textvariable</b></p> <p>Bir etiket widget'ında görüntülenen metni StringVar sınıfının bir kontrol değişkenine bağlamak için bu seçeneği o değişkene ayarlayın.</p>
24	<p><b>underline</b></p> <p>Bu seçeneği n olarak ayarlayarak metnin nth harfinin altında bir alt çizgi ( _ ) görüntüleyerek 0'dan sayabilirsiniz. Varsayılan alt çizgi = -1'dir, yani alt çizgi yoktur.</p>

25	<p><b>value</b></p> <p>Bir radiobutton nesnesi kullanıcı tarafından açıldığında, kontrol değişkeni geçerli değer seçeneğine ayarlanır. Kontrol değişkeni bir IntVar ise, gruptaki her bir radiobutton nesnesine farklı bir tam sayı değeri seçeneği verin. Kontrol değişkeni bir StringVar ise, her bir radiobutton nesnesine farklı bir string değeri seçeneği verin.</p>
26	<p><b>Variable</b></p> <p>Bu radiobutton nesnesinin gruptaki diğer radiobutton nesneleriyle paylaştığı kontrol değişkeni. Bu bir IntVar veya StringVar olabilir.</p>
27	<p><b>width</b></p> <p>Etiket içindeki karakterlerin genişliği (piksel değil!). Bu seçenek belirlenmemişse, etiket içeriğine uyacak şekilde boyutlandırılacaktır.</p>
28	<p><b>wraplength</b></p> <p>Bu seçeneği istenen sayıya ayarlayarak her satırdaki karakter sayısını sınırlayabilirsiniz. Varsayılan değer 0, satırların yalnızca yeni satırlarda kırılacağı anlamına gelir.</p>

### Metodlar

Radiobutton için en yaygın kullanılan metodlar şunlardır;

Sr.No.	Metod & Açıklama
1	<p><b>deselect()</b></p> <p>Radiobutton nesnesinin seçimini iptal eder.</p>
2	<p><b>flash()</b></p> <p>Aktif ve normal renkleri arasında radiobutton nesnesini birkaç kez yakıp söndürür, ancak başlattığı şekilde bırakır.</p>
3	<p><b>invoke()</b></p>

	Kullanıcı, durumunu değiştirmek için radiobutton nesnesine tıkladığında gerçekleşen aynı aksiyonları almak için bu yöntemi çağırabilirsiniz.
4	<b>select()</b> Radiobutton nesnesi seçilir.

Radiobutton kullanımına ait örnek kod Şekil – 12’da ve kodun çıktısı da Şekil – 12.1’de gösterilmiştir.

```

In [*]: #Radiobutton Kullanımı
import tkinter as tk

def sel():
    selection = "You selected the option " + str(var.get())
    label.config(text = selection)

win = tk.Tk()
var = tk.IntVar()

R1 = tk.Radiobutton(win, text="Option 1", variable=var, value=1, command=sel)
R1.pack( anchor = "center" )

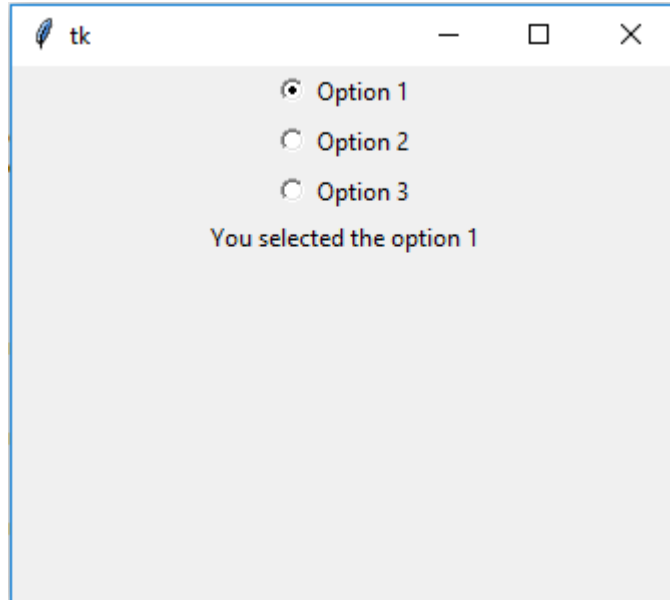
R2 = tk.Radiobutton(win, text="Option 2", variable=var, value=2, command=sel)
R2.pack( anchor = "center" )

R3 = tk.Radiobutton(win, text="Option 3", variable=var, value=3, command=sel)
R3.pack( anchor = "center" )

label = tk.Label(win)
label.pack()
win.mainloop()

```

Şekil – 12



Şekil – 12.1

➤ **Scale**

Bir kaydırıcı widget'ı sağlamak için kullanılır.

Sözdizimi (Syntax)

```
w = Scale ( master, option=value, ... )
```

Parametreler

- **master** : Ana pencereyi temsil eder
- **options** : Seçenekler listesini temsil eder. Bu seçenekler virgülle ayrılmış anahtar – değer çiftleri olarak kullanılır. En çok kullanılan seçeneklerin listesi aşağıda belirtilmiştir.

Sr.No.	Seçenek & Açıklama
1	<b>activebackground</b> Fare scale üzerinde olduğunda arka plan rengi.
2	<b>bg</b> Widget'ın çukurun dışındaki bölümlerinin arka plan rengi.
3	<b>bd</b> Çukur ve slider etrafındaki 3-d kenarlık genişliği. Varsayılan 2 pikseldir.
4	<b>command</b> Slider her hareket ettirildiğinde çağrılacak prosedür adını belirtir.
5	<b>cursor</b> Bu seçeneği bir imleç ismine (arrow, dot, vb.) ayarlarsanız, scale üzerinde olduğunda fare imleci o desene değişir.
6	<b>digits</b> Programınızın ölçek widget'ında gösterilen mevcut değeri okuduğu yol bir kontrol değişkenidir. Bir scale için kontrol değişkeni bir IntVar, bir DoubleVar (float) veya StringVar olabilir. Bir string değişkeni ise, basamak seçeneği, sayısal ölçek değeri bir string'e dönüştürüldüğünde kaç basamak kullanacağını belirtir.

7	<b>Font</b> Etiket ve ek açıklamalarda kullanılan yazı tipi.
8	<b>fg</b> Etiket ve ek açıklamalarda kullanılan metnin rengi.
9	<b>from_</b> Scale aralığının bir ucunu tanımlayan bir float veya tamsayı değeri.
10	<b>highlightbackground</b> Scale odaklanmadığı zaman odak rengini vurgular.
11	<b>highlightcolor</b> Scale odaklandığında ki odak rengi.
12	<b>label</b>  Bu seçeneği etiketin metnine ayarlayarak scale widget'ındaki bir etiketi görüntüleyebilirsiniz. Scale yataysa veya dikeyse sağ üst köşede, etiket sol üst köşede görünür. Varsayılan etiket değildir.
13	<b>length</b>  Scale parçacığının uzunluğu. Bu, scale yatay ise x boyutu veya dikey ise y boyutudur. Varsayılan 100 pikseldir.
14	<b>orient</b>  Scale'in x boyutu boyunca ilerlemesini istiyorsanız, orient=HORIZONTAL değerini veya y eksenine paralel olarak çalışmasını istiyorsanız orient=VERTICAL değerini ayarlamalısınız. Varsayılan yataydır.
15	<b>Relief</b>  Etiketin etrafında dekoratif bir sınırın görünümünü belirtir. Varsayılan FLAT'tır.
16	<b>repeatdelay</b>

	<p>Bu seçenek, slider tekrar tekrar hareket etmeye başlamadan önce 1 nolu butonun çukurda ne kadar bekletileceğini belirtir. Varsayılan repeatdelay=300 ve birimler milisaniyedir.</p>
17	<p><b>resolution</b></p> <p>Normalde, kullanıcı yalnızca tüm birimlerde scale değiştirebilir. Scale değerinin en küçük artışı değiştirmek için bu seçeneği başka bir değere ayarlayın. Örneğin, = - 1.0 ve = 1.0 arasında bir değer varsa ve çözünürlük = 0,5'i ayarlarsanız, ölçek 5 olası değere sahip olacaktır: -1.0, -0.5, 0.0, +0.5 ve +1.0.</p>
18	<p><b>showvalue</b></p> <p>Normalde, scale'in geçerli değeri slider tarafından metin biçiminde görüntülenir (yatay ölçekler için, dikey ölçekler için sola). Bu etiketi bastırmak için bu seçeneği 0 olarak ayarlayın.</p>
19	<p><b>sliderlength</b></p> <p>Normalde slider, scale uzunluğu boyunca 30 pikseldir. sliderlength seçeneğini istediğiniz uzunluğa ayarlayarak bu uzunluğu değiştirebilirsiniz.</p>
20	<p><b>state</b></p> <p>Normal olarak, scale widget'ları fare olaylarına yanıt verir ve odaklandıklarında klavye olayları da olur. Widget'ı devre dışı bırakmak için state = DISABLED yapılır.</p>
21	<p><b>takefocus</b></p> <p>Normalde odak, scale widget'ları arasında değişecektir. Bu davranışı istemiyorsanız, bu seçeneği 0 olarak ayarlayın.</p>
22	<p><b>tickinterval</b></p> <p>Periyodik scale değerlerini görüntülemek için, bu seçeneği bir sayıya ayarlayın ve bu değer katlarında işaretler görüntülenir. Örneğin, eğer from_ = 0.0, = 1.0 ve tickinterval = 0.25 ise, scale boyunca 0.0, 0.25, 0.50, 0.75 ve 1.00 değerlerinde etiketler gösterilecektir. Bu etiketler yataysa ölçeğin altında, dikey ise solunda görünür. Varsayılan, işaretlerin görüntülenmesini engelleyen 0'dır.</p>



23	<p><b>to</b></p> <p>Scale'in aralığının bir ucunu tanımlayan bir float veya integer değeri; Diğer uç yukarıda tartışılan from_ seçeneği ile tanımlanır. Bu değer, from_ değerinden daha büyük veya daha küçük olabilir. Dikey scale'lar için değer ölçeğin altını tanımlar; yatay scale'lar için sağ son.</p>
24	<p><b>troughcolor</b></p> <p>Çukurun rengi.</p>
25	<p><b>variable</b></p> <p>Varsa bu scale için kontrol değişkeni. Kontrol değişkenleri IntVar, DoubleVar (float) veya StringVar sınıfından olabilir. İkinci durumda, sayısal değer bir string'e dönüştürülecektir.</p>
26	<p><b>width</b></p> <p>Widget'ın çukur kısmının genişliği. Bu, dikey scale'lar için x boyutudur ve ölçeğin orient = HORIZONTAL olması durumunda y boyutu. Varsayılan 15 pikseldir.</p>

### Metodlar

Scale için en yaygın kullanılan metodlar şunlardır;

Sr.No.	Metod & Açıklama
1	<p><b>get()</b></p> <p>Bu yöntem, scale'in geçerli değerini döndürür.</p>
2	<p><b>set ( value )</b></p> <p>Scale'in değerini ayarlar.</p>

Scale kullanımına ait örnek kod Şekil – 13’da ve kodun çıktısı da Şekil – 13.1’de gösterilmiştir.

```
In [21]: #Scale Kullanımı
import tkinter as tk

def sel():
    selection = "Değer = " + str(var.get())
    label.config(text = selection)

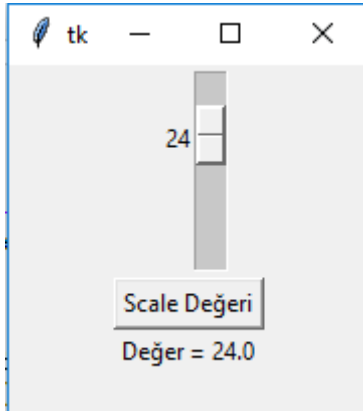
win = tk.Tk()
var = tk.DoubleVar()
scale = tk.Scale(win, variable = var )
scale.pack(anchor="center")

button = tk.Button(win, text="Scale Değeri", command=sel)
button.pack(anchor="center")

label = tk.Label(win)
label.pack()

win.mainloop()
```

Şekil – 13



Şekil – 13.1

### ➤ Scrollbar

Liste kutuları gibi çeşitli widget'lara kaydırma özelliği eklemek için kullanılır.

#### Sözdizimi (Syntax)

```
w = Scrollbar ( master, option=value, ... )
```

#### Parametreler

- **master** : Ana pencereyi temsil eder
- **options** : Seçenekler listesini temsil eder. Bu seçenekler virgülle ayrılmış anahtar – değer çiftleri olarak kullanılır. En çok kullanılan seçeneklerin listesi aşağıda belirtilmiştir.

Sr.No.	Seenek & Açıklama
1	<b>activebackground</b> Fare üstündeyken slider'ın ve ok uçlarının rengi.
2	<b>bg</b> Fare üzerlerinde değilken slider'ın ve ok uçlarının rengi.
3	<b>bd</b> Çukurun tüm çevresi etrafında 3-d kenarlık genişliği ve aynı zamanda ok uçları ve slider üzerindeki 3-d etkilerinin genişliği. Varsayılan, çukurun etrafında bir sınır ve ok uçları ve slider etrafındaki 2 piksellik bir sınırdır.
4	<b>command</b> Kaydırma çubuğu hareket ettirildiğinde çağrılacak prosedür adı.
5	<b>cursor</b> Fare kaydırma çubuğu üzerinde olduğunda görüntülenen imleç.
6	<b>elementborderwidth</b> Ok uçları ve slider etrafındaki kenarların genişliği. Varsayılan, borderwidth seçeneğinin değerini kullanmak anlamına gelen elementborderwidth = -1'dir.
7	<b>highlightbackground</b> Kaydırma çubuğunun odaklanmadığı zaman odak rengini belirtir.
8	<b>highlightcolor</b> Kaydırma çubuğu odağa sahipken odak rengini belirtir.
9	<b>highlightthickness</b> Odağın kalınlığını belirtir. Varsayılan 1'dir. Odak vurgulama ekranını bastırmak için 0'a ayarlayın.

10	<p><b>jump</b></p> <p>Bu seçenek, kullanıcı slider'ı sürüklediğinde ne olacağını kontrol eder. Normalde (atlama = 0), slider'ın her küçük sürüklemesi, komut çağrısının çağırılmasına neden olur. Bu seçeneği 1 olarak ayarlarsanız, kullanıcı fare düğmesini serbest bırakana kadar fonksiyon çağırılmaz.</p>
11	<p><b>orient</b></p> <p>Yatay kaydırma çubuğu için orient = HORIZONTAL, dikey kaydırma çubuğu için orient = VERTICAL ayarlayınız.</p>
12	<p><b>repeatdelay</b></p> <p>Bu seçenek, slider'ın tekrar tekrar bu yönde hareket etmeden önce, 1 nolu buton üzerinde aşağıya ne kadar basılması gerektiğini kontrol eder. Varsayılan repeatdelay = 300'dür ve birimler milisaniyedir.</p>
13	<p><b>repeatinterval</b></p> <p>Tekrar aralığı</p>
14	<p><b>takefocus</b></p> <p>Normalde, odağı bir kaydırma çubuğu widget'ından sekme yapabilirsiniz. Bu davranışı istemiyorsanız takefocus = 0 ögesini ayarlayın.</p>
15	<p><b>troughcolor</b></p> <p>Çukurun rengi.</p>
16	<p><b>width</b></p> <p>Kaydırma çubuğunun genişliği (yatay ise y boyutu ve dikey ise x boyutu). Varsayılan 16'dır.</p>

### Metodlar

Scrollbar için en yaygın kullanılan metodlar şunlardır;

Sr.No.	Metod & Açıklama
1	<b>get()</b>  Slider'ın geçerli konumunu açıklayan iki sayıyı (a, b) döndürür. Bir değer, sırasıyla yatay ve dikey kaydırma çubukları için kaydırıcının sol veya üst kenarının konumunu verir; b değeri, sağ veya alt kenarın konumunu verir.
2	<b>set ( first, last )</b>  Bir kaydırma çubuğunu başka bir w widget'a bağlamak için w'nin xscrollcommand veya yscrollcommand ögesini kaydırma çubuğunun set() yöntemine ayarlayın. Argümanlar, get() yöntemiyle döndürülen değerler ile aynı anlama sahiptir.

Scrollbar kullanımına ait örnek kod Şekil – 14'da ve kodun çıktısı da Şekil – 14.1'de gösterilmiştir.

```
In [27]: #Scrollbar Kullanımı
import tkinter as tk

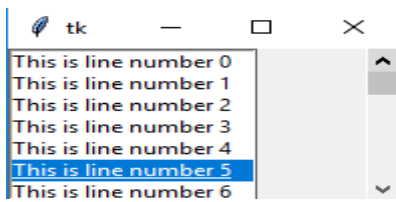
win = tk.Tk()
scrollbar = tk.Scrollbar(win)
scrollbar.pack( side = "right", fill = "both" )

mylist = tk.Listbox(win, yscrollcommand = scrollbar.set )
for line in range(100):
    mylist.insert("end", "This is line number " + str(line))

mylist.pack( side = "left", fill = "both" )
scrollbar.config( command = mylist.yview )

win.mainloop()
```

Şekil – 14



Şekil – 14.1

➤ **Text**

Metni birden çok satırda görüntülemek için kullanılır.

Sözdizimi (Syntax)

```
w = Text ( master, option=value, ... )
```

Parametreler

▪ **master** : Ana pencereyi temsil eder  
**options** : Seçenekler listesini temsil eder. Bu seçenekler virgülle ayrılmış anahtar – değer çiftleri olarak kullanılır. En çok kullanılan seçeneklerin listesi aşağıda belirtilmiştir.

Sr.No.	Seçenek & Açıklama
1	<b>bg</b> Varsayılan arka plan rengi.
2	<b>bd</b> Metin widget'ının etrafındaki kenarlık genişliği. Varsayılan 2 pikseldir.
3	<b>cursor</b> Fare, metin parçacığının üzerindeyken görünecek imleç.
4	<b>exportselection</b> Normal olarak, bir metin widget'ında seçilen metin, pencere yöneticisinde seçim olarak dışa aktarılır. Bu davranışı istemiyorsanız exportselection = 0 değerini ayarlayın.
5	<b>font</b> Widget'a eklenen metin için varsayılan yazı tipi.
6	<b>fg</b> Widget içindeki metin (ve bitmapler) için kullanılan renk. Etiketli bölgeler için rengi değiştirebilirsiniz; Bu seçenek sadece varsayılan değerdir.
7	<b>height</b>

	Geçerli yazı tipi boyutuna göre ölçülen satırdaki (piksel değil!) satırın yüksekliği.
8	<b>highlightbackground</b> Metin parçacığının odaklanmadığı zaman ki odak rengini belirtir.
9	<b>highlightcolor</b> Metin parçacığı odağa sahip olduğunda ki odak rengi.
10	<b>highlightthickness</b> Odağın kalınlığını belirtir. Varsayılan 1'dir. Odak vurgulamasını görüntülemek için highlightthickness = 0 değerini ayarlayın.
11	<b>insertbackground</b> Ekleme imlecinin rengi. Varsayılan siyah.
12	<b>insertborderwidth</b> Ekleme imlecinin etrafındaki 3 boyutlu sınırın boyutu. Varsayılan 0'dır.
13	<b>insertofftime</b> Yanıp sönme döngüsü sırasında ekleme imlecinin kapalı olduğu milisaniye sayısı. Yanıp sönme özelliğini kapatmak için bu seçeneği sıfıra ayarlayın. Varsayılan 300'dür.
14	<b>insertontime</b> Ekleme imlecinin yanıp sönme döngüsü sırasında açık olduğu milisaniye sayısı. Varsayılan 600'dür.
15	<b>insertwidth</b> Ekleme imlecinin genişliği (yüksekliği, çizgisindeki en uzun öğeye göre belirlenir). Varsayılan 2 pikseldir.
16	<b>padx</b> Metin alanının soluna ve sağına eklenen iç dolgu boyutu. Varsayılan bir

	pikseldir.
17	<p><b>pady</b></p> <p>Metin alanının üstünde ve altında eklenen iç dolgu boyutu. Varsayılan bir pikseldir.</p>
18	<p><b>relief</b></p> <p>Metin parçacığının 3 boyutlu görünüşü. Varsayılan relief = SUNKEN.</p>
19	<p><b>selectbackground</b></p> <p>Seçilen metni görüntülemek için kullanılacak arka plan rengi.</p>
20	<p><b>selectborderwidth</b></p> <p>Seçilen metnin etrafındaki sınırın genişliği.</p>
21	<p><b>spacing1</b></p> <p>Bu seçenek, her bir metin satırının üzerine ne kadar fazla dikey boşluk yerleştirildiğini belirtir. Bir satır sararsa, bu alan yalnızca ekranda kapladığı ilk satırdan önce eklenir. Varsayılan 0'dır.</p>
22	<p><b>spacing2</b></p> <p>Bu seçenek, mantıksal bir satır sararken görüntülenen metin satırları arasında ne kadar ek dikey boşluk ekleneceğini belirtir. Varsayılan 0'dır.</p>
23	<p><b>spacing3</b></p> <p>Bu seçenek, her bir metin satırının altına ne kadar ek dikey boşluk eklendiğini belirtir. Bir satır sararsa, bu alan yalnızca ekranda kapladığı son satırdan sonra eklenir. Varsayılan 0'dır.</p>
24	<p><b>state</b></p> <p>Normal olarak, metin widget'ları klavye ve fare olaylarına yanıt verir; Bu davranışı almak için state = NORMAL ayarlayın. state = DISABLED değerini ayarlarsanız, metin widget'ı yanıt vermez ve içeriklerini programsal olarak</p>



	değiştiremezsiniz.
25	<b>tabs</b> Bu seçenek, sekme karakterlerinin metni nasıl konumlandığını kontrol eder.
26	<b>width</b> Widget'ın karakterlerde genişliği (piksel değil!), geçerli yazı tipi boyutuna göre ölçülür.
27	<b>wrap</b> Bu seçenek çok geniş olan satırların görünümünü kontrol eder. wrap = WORD ögesini ayarlayın ve bu sığacak olan son sözcükten sonra satırı kesecektir. Varsayılan davranışla wrap = CHAR, çok uzun olan herhangi bir satır herhangi bir karakterde kırılır.
28	<b>xscrollcommand</b> Metin widget'ını yatay kaydırılabilir yapmak için, bu seçeneği yatay kaydırma çubuğunun set () yöntemine ayarlayın.
29	<b>yscrollcommand</b> Metin widget'ını dikey olarak kaydırılabilir yapmak için, bu seçeneği dikey kaydırma çubuğunun set () yöntemine ayarlayın.

### Metodlar

Text için en yaygın kullanılan metodlar şunlardır;

Sr.No.	Metod & Açıklama
1	<b>delete(startindex [,endindex])</b> Bu yöntem belirli bir karakteri veya bir metin aralığını siler.
2	<b>get(startindex [,endindex])</b> Bu yöntem belirli bir karakter veya bir metin aralığı döndürür.

3	<b>index(index)</b> Verilen index'e göre index'in kesin deęerini döndürür.
4	<b>insert(index [,string]...)</b> Bu yöntem belirtilen index konumuna verilen string'leri ekler.
5	<b>see(index)</b> indeks pozisyonunda bulunan metin görünür olması durumunda bu yöntem true döndürür.

Metin widget'ları üç farklı yardımcı yapıyı destekler: İşaretler, Sekmeler ve Indexler -

İşaretler, belirli bir metindeki iki karakter arasındaki konumları işaretlemek için kullanılır. İşaretleri kullanırken aşağıdaki yöntemlere sahibiz –

Sr.No.	Metod & Açıklama
1	<b>index(mark)</b> Belirli bir işaretin satır ve sütun konumunu döndürür.
2	<b>mark_gravity(mark [,gravity])</b> Verilen işaretin ağırlığını döndürür. İkinci argüman sağlanırsa, verilen işaret için ağırlığı ayarlanır.
3	<b>mark_names()</b> Metin widget'ındaki tüm işaretleri döndürür.
4	<b>mark_set(mark, index)</b> Verilen işarete yeni bir pozisyon bildirir.
5	<b>mark_unset(mark)</b> Belirtilen işaretini Metin widget'ından kaldırır.

Etiketler, belirli metin alanlarının ekran ayarlarını deęiřtirme görevini kolaylařtıran metin bölgelerine isimleri iliřkilendirmek için kullanılır. Etiketler ayrıca fonksiyon çağırmaıı belirli metin aralıklarına bağlamak için kullanılır.

Ařaęıdaki sekmeleri işlemek için kullanılabilir yöntemler –

Sr.No.	Metod & Açıklama
1	<b>tag_add(tagname, startindex[,endindex] ...)</b> Bu yöntem, startindex tarafından tanımlanan pozisyonu veya startindex ve endindex pozisyonları ile sınırlanmış bir aralıęı etiketler.
2	<b>tag_config</b> Bu özellięi, gerekçelendirme (orta, sol veya saę), sekmeleri (bu özellik Metin widget'ı sekmesinin özellięi ile aynı işlevsellięe sahiptir) ve alt çizgi (etiketli metnin altını çizmek için kullanılır) içeren etiket özelliklerini yapılandırmak için kullanabilirsiniz.
3	<b>tag_delete(tagname)</b> Bu yöntem, belirli bir etiketi silmek ve kaldırmak için kullanılır.
4	<b>tag_remove(tagname [,startindex[,endindex]] ...)</b> Bu yöntemi uyguladıktan sonra, verilen etiket gerçek etiket tanımını silmeden sağlanan alandan kaldırılır.

Text kullanımına ait örnek kod Şekil – 15'da ve kodun çıktısı da Şekil – 15.1'de gösterilmiştir.

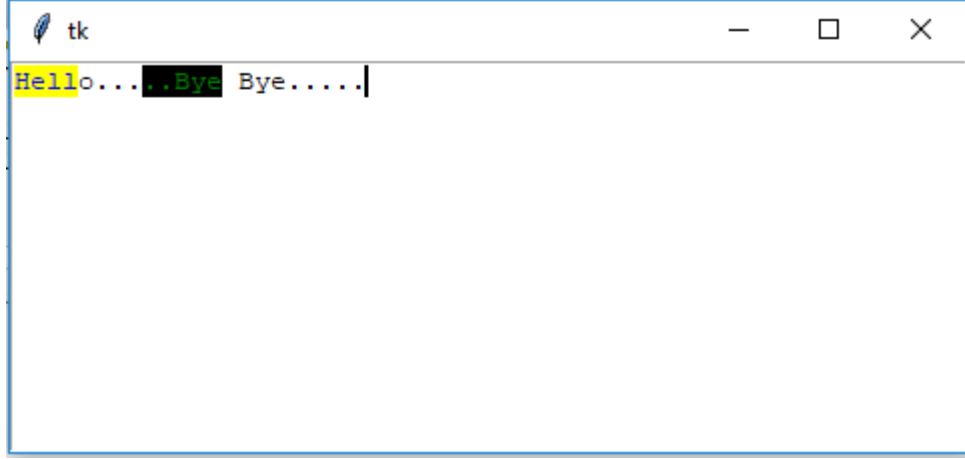
```
In [31]: #Text Kullanımı
import tkinter as tk

def onclick():
    pass

win = tk.Tk()
text = tk.Text(win)
text.insert("insert", "Hello.....")
text.insert("end", "Bye Bye.....")
text.pack()

text.tag_add("here", "1.0", "1.4")
text.tag_add("start", "1.8", "1.13")
text.tag_config("here", background="yellow", foreground="blue")
text.tag_config("start", background="black", foreground="green")
win.mainloop()
```

Şekil – 15



Şekil – 15.1

#### ➤ Toplevel

Toplevel widget'ları, pencere yöneticisi tarafından doğrudan yönetilen pencereler olarak çalışır. Bunların üzerinde mutlaka bir Toplevel ögesi bulunmak zorunda değildir. Uygulamanız herhangi bir sayıda Toplevel pencereyi kullanabilir.

#### Sözdizimi (Syntax)

```
w = Toplevel ( option=value, ... )
```

#### Parametreler

- **options** : Bu widget için en çok kullanılan seçeneklerin listesi aşağıdadır. Bu seçenekler, virgülle ayrılmış anahtar / değer çiftleri olarak kullanılabilir.

Sr.No.	Seçenek & Açıklama
1	<b>bg</b> Pencerenin arka plan rengi.
2	<b>bd</b> Piksel cinsinden kenarlık genişliği; varsayılan 0'dır.
3	<b>cursor</b> Fare bu pencerede olduğunda görüntülenen imleç.
4	<b>class_</b> Normal olarak, bir metin widget'ında seçilen metin, pencere yöneticisinde

	seçim olarak dışa aktarılır. Bu davranışı istemiyorsanız exportselection = 0 değerini ayarlayın.
5	<b>font</b> Widget'a eklenen metin için varsayılan yazı tipi.
6	<b>fg</b> Widget içindeki metin (ve bitmapler) için kullanılan renk. Etiketli bölgeler için rengi değiştirebilirsiniz; Bu seçenek sadece varsayılan değerdir.
7	<b>height</b> Pencere yüksekliği.
8	<b>relief</b> Normalde, Toplevel bir pencerenin etrafında 3-d sınırları olmaz. Gölgeli bir kenarlık elde etmek için, bd seçeneğini varsayılan sıfır değerinden daha büyük olarak ayarlayın ve kabartma seçeneğini sabitlerden birine ayarlayın.
9	<b>width</b> Pencerenin istenilen genişliği.

### Metodlar

Toplevel nesneleri bu yöntemlere sahiptir;

Sr.No.	Metod & Açıklama
1	<b>deiconify()</b> Simgeyi veya çizim yöntemlerini kullandıktan sonra pencereyi görüntüler.
2	<b>frame()</b> Sisteme özgü bir pencere tanıtıcısı döndürür.
3	<b>group(window)</b>

	Pencereyi, verilen pencere tarafından uygulanan pencere grubuna ekler.
4	<b>iconify()</b> Pencereyi yok etmeden bir simgeye dönüştürür.
5	<b>protocol(name, function)</b> Verilen protokol için çağrılacak fonksiyon işlevi kaydeder.
6	<b>state()</b> Pencerenin mevcut durumunu döndürür. Olası değerler normal, iconic, withdrawn ve icon.
7	<b>transient([master])</b> Hiçbir argüman verilmediğinde, pencereyi verilen master veya pencerenin ebeveyni için geçici (geçici) bir pencereye döndürür.
8	<b>withdraw()</b> Pencereyi ekrandan yok etmeden kaldırır. Gizler.
9	<b>maxsize(width, height)</b> Bu pencerenin maksimum boyutunu tanımlar.
10	<b>minsize(width, height)</b> Bu pencerenin minimum boyutunu tanımlar.
11	<b>positionfrom(who)</b> Pozisyon kontrolünü tanımlar.
12	<b>resizable(width, height)</b> Pencerenin yeniden boyutlandırılabilceğini kontrol eden yeniden boyutlandırma bayraklarını tanımlar.
13	<b>sizefrom(who)</b>

	Boyut kontrolünü tanımlar.
14	<b>title(string)</b> Pencere başlığını tanımlar.

Toplevel kullanımına ait örnek kod Şekil – 16’da ve kodun çıktısı da Şekil – 16.1 ve Şekil – 16.2’de gösterilmiştir.

```
from tkinter import *

class AnaForm():
    def __init__(self, master):
        self.master = master
        self.master.geometry("400x200+100+200")
        self.master.title("Ana Form")

        self.btnHesaplamaFormu = Button(self.master, text="Hesaplama", fg="blue", command=self.gosterHesFrn).grid(row=1, column=0)
        self.btnCikis = Button(self.master, text="Çıkış", fg="blue", command=self.cikis).grid(row=1, column=1)

    def gosterHesFrn(self):
        winHesaplamaFormu = Toplevel(self.master)
        guiHesaplamaFormu = HesaplamaFormu(winHesaplamaFormu)

    def cikis(self):
        self.master.destroy()

class HesaplamaFormu():
    def __init__(self, master):
        self.calisilan_saat = DoubleVar()
        self.saat_ucreti = DoubleVar()

        self.master = master
        self.master.geometry("500x200+100+200")
        self.master.title("Hesaplama Formu")

        self.label1 = Label(self.master, text="Hesaplama Formu GUI", fg="red").grid(row=0, column=2)
        self.label2 = Label(self.master, text="Saatlik ücretiniz").grid(row=3, column=0)
        self.label3 = Label(self.master, text="Çalışma süresiniz (saati)").grid(row=4, column=0)

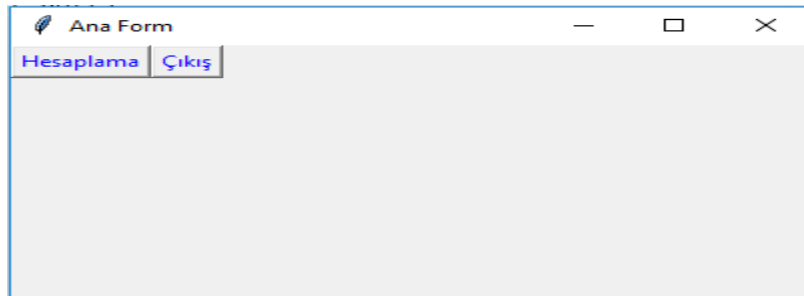
        self.entry_saat_ucreti = Entry(self.master, textvariable=self.saat_ucreti).grid(row=3, column=2)
        self.entry_calisilan_saat = Entry(self.master, textvariable=self.calisilan_saat).grid(row=4, column=2)
        self.btnMaasHesapla = Button(self.master, text="Maaş Hesapla", fg="blue", command=self.hesaplaMaas).grid(row=5, column=2)
        self.btnGeri = Button(self.master, text="Geri Dön", fg="blue", command=self.geri).grid(row=5, column=3)

    def hesaplaMaas(self):
        saat = self.calisilan_saat.get()
        print(saat)
        saat_ucret = self.saat_ucreti.get()
        maas = saat * saat_ucret
        print(maas)
        self.labelResult = Label(self.master, text="Maaşınız : %.2f TL" % maas).grid(row=7, column=2)

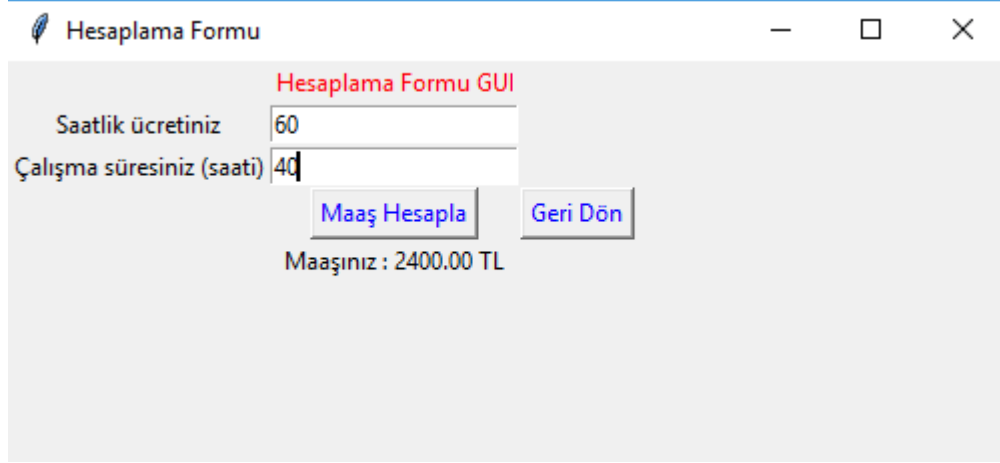
    def geri(self):
        self.master.destroy()

def main():
    win = Tk()
    guiAnaFormu = AnaForm(win)
```

Şekil – 16



Şekil – 16.1



Şekil – 16.2

### ➤ Spinbox

Sabit sayıda sayıdan seçim yapmak için kullanılabilen standart Tkinter Giriş widget'ının bir çeşididir.

#### Sözdizimi (Syntax)

```
w = Spinbox ( master, option=value, ... )
```

#### Parametreler

- **master** : Ana pencereyi temsil eder
- **options** : Seçenekler listesini temsil eder. Bu seçenekler virgülle ayrılmış anahtar – değer çiftleri olarak kullanılır. En çok kullanılan seçeneklerin listesi aşağıda belirtilmiştir.

Sr.No.	Seçenek & Açıklama
1	<b>activebackground</b> Fare üstündeyken slider ve ok uçlarının rengi.
2	<b>bg</b> Fare üzerlerinde değilken slider'ın ve ok uçlarının rengi.
3	<b>bd</b> Çukurun tüm çevresi etrafında 3-d kenarlık genişliği ve aynı zamanda ok uçları ve slider üzerindeki 3-d etkilerinin genişliği. Varsayılan, çukurun etrafında bir sınır ve ok uçları ve slider etrafındaki 2 piksellik bir sınırdır.
4	<b>command</b>



	Kaydırma çubuğu hareket ettirildiğinde çağrılacak fonksiyon adı.
5	<b>cursor</b> Fare kaydırma çubuğu üzerinde olduğunda görüntülenen imleç.
6	<b>disabledbackground</b> Widget devre dışı bırakıldığında kullanılacak arka plan rengi.
7	<b>disabledforeground</b> Widget devre dışı bırakıldığında kullanılacak metin rengi.
8	<b>fg</b> Metin rengi.
9	<b>font</b> Bu widget'ta kullanılacak yazı tipi.
10	<b>format</b> String formatı. Varsayılan değer yok.
11	<b>from_</b> Minimum değer. Spinbox aralığını sınırlamak için birlikte kullanılır.
12	<b>justify</b> Varsayılan LEFT
13	<b>relief</b> Varsayılan SUNKEN.
14	<b>repeatdelay</b> Repeatinterval ile birlikte, bu seçenek düğmeyi otomatik tekrarlamayı kontrol eder. Her iki değer de milisaniyede verilir.

15	<b>repeatinterval</b> repeatdelay bakınız.
16	<b>state</b> NORMAL, DISABLED veya "readonly" öğelerinden biri. Varsayılan NORMAL'dir.
17	<b>textvariable</b> Varsayılan değer yok.
18	<b>To</b> from konusuna bakınız.
19	<b>validate</b> Doğrulama modu. Varsayılan NONE'dir.
20	<b>validatecommand</b> Doğrulama fonksiyonu adı. Varsayılan değer yok.
21	<b>values</b>  Bu widget için geçerli değerler içeren bir tuple. from/to/increment'i geçersiz kılar.
22	<b>vcmd</b> Validatecommand ile aynı.
23	<b>width</b> Widget genişliği, karakter birimlerinde. Varsayılan 20'dir.
24	<b>wrap</b> Doğruysa, yukarı ve aşağı düğmeleri etrafına sarılacaktır.
25	<b>xscrollcommand</b>

Bir spinbox alanını yatay bir kaydırma çubuğuna bağlamak için kullanılır. Bu seçenek, ilgili kaydırma çubuğunun ayarlanan yöntemine ayarlanmalıdır.

### Metodlar

Spinbox nesneleri bu yöntemlere sahiptir;

Sr.No.	Metod & Açıklama
1	<b>delete(startindex [,endindex])</b> Bu yöntem belirli bir karakteri veya bir metin aralığını siler.
2	<b>get(startindex [,endindex])</b> Bu yöntem belirli bir karakter veya bir metin aralığı döndürür.
3	<b>identify(x, y)</b> Verilen konumdaki widget ögesini tanımlar.
4	<b>index(index)</b> Belirtilen index'e dayalı bir index'in mutlak değerini döndürür.
5	<b>insert(index [,string]...)</b> Bu yöntem belirtilen index konumuna string'leri ekler.
6	<b>invoke(element)</b> Bir spinbox düğmesi çağırır.

Spinbox kullanımına ait örnek kod Şekil – 17'da ve kodun çıktısı da Şekil – 17.1'de gösterilmiştir.

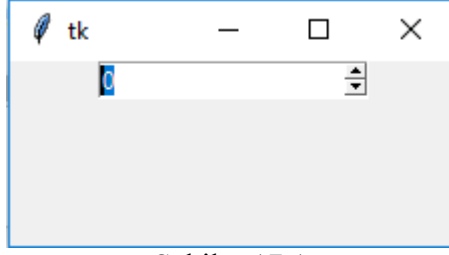
```
#Spinbox Kullanım
import tkinter as tk

win = tk.Tk()

sb = tk.Spinbox(win, from_=0, to=10)
sb.pack()

win.mainloop()
```

Şekil – 17



Şekil – 17.1

➤ **PanedWindow**

Yatay veya dikey olarak düzenlenmiş herhangi bir sayıda bölmeyi içerebilen bir kapsayıcı pencere ögesidir.

Sözdizimi (Syntax)

```
w = PanedWindow ( master, option=value, ... )
```

Parametreler

- **master** : Ana pencereyi temsil eder
- **options** : Seçenekler listesini temsil eder. Bu seçenekler virgülle ayrılmış anahtar – değer çiftleri olarak kullanılır. En çok kullanılan seçeneklerin listesi aşağıda belirtilmiştir.

Sr.No.	Seçenek & Açıklama
1	<b>bg</b> Fare üzerlerinde değilken slider'ın ve ok uçlarının rengi.
2	<b>bd</b> Çukurun tüm çevresi etrafında 3-d kenarlık genişliği ve aynı zamanda ok uçları ve slider üzerindeki 3-d etkilerinin genişliği. Varsayılan, çukurun etrafında bir sınır ve ok uçları ve slider etrafındaki 2 piksellik bir sınırdır.
3	<b>borderwidth</b> Varsayılan 2'dir.
4	<b>cursor</b> Fare pencerenin üzerindeyken görünen imleç.
5	<b>handlepad</b> Varsayılan 8'dir.

6	<b>handlesize</b> Varsayılan 8'dir.
7	<b>height</b> Varsayılan değer yok.
8	<b>orient</b> Varsayılan HORIZONTAL.
9	<b>relief</b> Varsayılan FLAT.
10	<b>sashcursor</b> Varsayılan değer yok.
11	<b>sashrelief</b> Varsayılan RAISED.
12	<b>sashwidth</b> Varsayılan 2.
13	<b>showhandle</b> Varsayılan değer yok.
14	<b>width</b> Varsayılan değer yok.

### Metodlar

PanedWindow nesneleri bu yöntemlere sahiptir;

Sr.No.	Metod & Açıklama
1	<b>add(child, options)</b> Paned penceresine bir çocuk penceresi ekler.
2	<b>get(startindex [,endindex])</b> Bu yöntem belirli bir karakter veya bir metin aralığı döndürür.
3	<b>config(options)</b> Bir veya daha fazla widget seçeneğini değiştirir. Hiçbir seçenek belirtilmediyse, yöntem tüm geçerli seçenek değerlerini içeren bir sözlük döndürür.

PanedWindow kullanımına ait örnek kod Şekil – 18’da ve kodun çıktısı da Şekil – 18.1’de gösterilmiştir.

```
#PanedWindow Kullanımı
import tkinter as tk

m1 = tk.PanedWindow(bg="blue")
m1.pack(fill="both", expand=1)

left = tk.Label(m1, text="left pane")
m1.add(left)

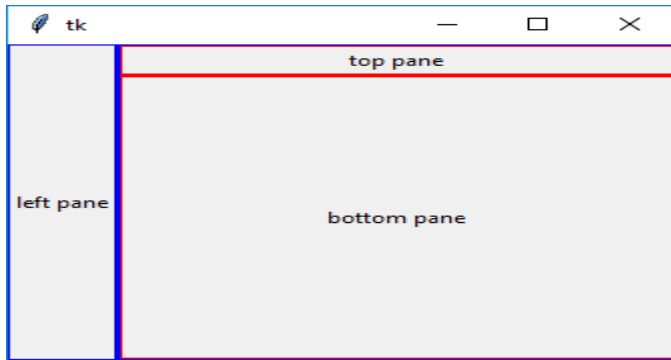
m2 = tk.PanedWindow(m1, bg="red", orient="vertical")
m1.add(m2)

top = tk.Label(m2, text="top pane")
m2.add(top)

bottom = tk.Label(m2, text="bottom pane")
m2.add(bottom)

tk.mainloop()
```

Şekil – 18



Şekil – 18.1

➤ **LabelFrame**

Basit konteyner widget'ıdır. Karmaşık pencere düzenleri için bir konteyner olarak kullanılır.

Sözdizimi (Syntax)

```
w = LabelFrame ( master, option=value, ... )
```

Parametreler

- **master** : Ana pencereyi temsil eder
- **options** : Seçenekler listesini temsil eder. Bu seçenekler virgülle ayrılmış anahtar – değer çiftleri olarak kullanılır. En çok kullanılan seçeneklerin listesi aşağıda belirtilmiştir.

Sr.No.	Seçenek & Açıklama
1	<b>bg</b> Etiket ve göstergenin arkasında görüntülenen normal arka plan rengi.
2	<b>bd</b> Gösterge etrafındaki sınırın boyutu. Varsayılan 2 pikseldir.
3	<b>cursor</b> Bu seçeneği bir imleç ismine (arrow, dot vb.) ayarlarsanız, fare imleci, kontrol tuşunun üzerindeyken o desene değişir.
4	<b>font</b> Bu widget'ta kullanılacak yazı tipi.
5	<b>height</b> Pencere için istenen yüksekliği belirtir.
6	<b>labelAnchor</b> Etiketi nereye yerleştireceğinizi belirtir.
7	<b>highlightbackground</b> Çerçevenin odaklanmadığı zaman odak rengini belirtir.

8	<b>highlightcolor</b> Çerçeve odakta olduğunda odakta gösterilen renk vurgulanır.
9	<b>highlightthickness</b> Odak vurgulama kalınlığı.
10	<b>relief</b> Varsayılan değer olan relief = FLAT ile, kontrol düğmesi arka planında göze çarpmaz. Bu seçeneği diğer stillerden birine ayarlayabilirsiniz.
11	<b>text</b> Widget içinde görüntülenecek bir metni belirtir.
12	<b>width</b> Pencere için istenen genişliği belirtir.

LabelFrame kullanımına ait örnek kod Şekil – 19’da ve kodun çıktısı da Şekil – 19.1’de gösterilmiştir.

```
#LabelFrame Kullanımı
import tkinter as tk

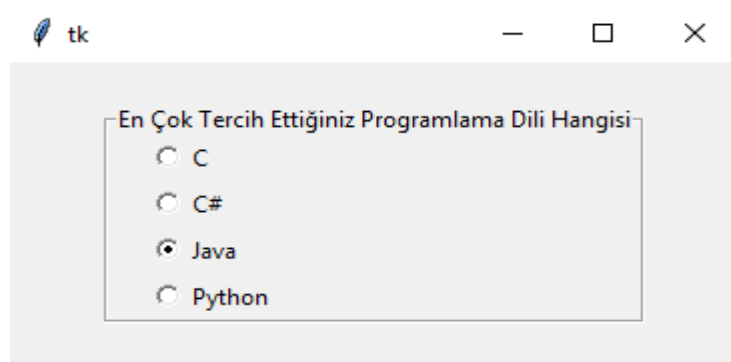
win = tk.Tk()
var = tk.IntVar()

labelframe = tk.LabelFrame(win, text="En Çok Tercih Ettiğiniz Programlama Dili Hangisi", labelanchor="n")
labelframe.pack(expand="yes")

rb1 = tk.Radiobutton(labelframe, text="C", padx = 20, variable=var, value=1).pack(anchor=tk.W)
rb2 = tk.Radiobutton(labelframe, text="C#", padx = 20, variable=var, value=2).pack(anchor=tk.W)
rb3 = tk.Radiobutton(labelframe, text="Java", padx = 20, variable=var, value=3).pack(anchor=tk.W)
rb4 = tk.Radiobutton(labelframe, text="Python", padx = 20, variable=var, value=4).pack(anchor=tk.W)

win.mainloop()
```

Şekil – 19



Şekil – 19.1



### ➤ **messagebox**

Bu modül, uygulamalarınızdaki mesaj kutularını görüntülemek için kullanılır. Ayrıca, uygun bir mesajı görüntülemek için kullanabileceğiniz bir dizi işlev sunar.

Bu işlevlerden bazıları showinfo, showwarning, showerror, askquestion, askokcancel, askyesno, and askretryignore.

#### Sözdizimi (Syntax)

```
w = messagebox.FunctionName ( title, message [, options] )
```

#### Parametreler

- **FunctionName** : Uygun mesaj kutusu işlevinin adıdır.
- **title** : Başlık çubuğunda görüntülenecek metin.
- **message** : Mesaj olarak görüntülenecek metin.
- **options** : Seçenekler, standart bir mesaj kutusunu uyarlamak için kullanabileceğiniz alternatif seçimlerdir. Kullanabileceğiniz bazı seçenekler varsayılan ve parent'tır. Varsayılan seçenek, mesaj kutusunda ABORT, RETRY veya IGNORE gibi varsayılan düğmeyi belirtmek için kullanılır. Parent seçeneği, mesaj kutusunun görüntüleneceği pencereyi belirtmek için kullanılır.

Diyalog kutusu ile aşağıdaki fonksiyonlardan birini kullanabilirsiniz.

- showinfo()
- showwarning()
- showerror ()
- askquestion()
- askokcancel()
- askyesno ()
- askretrycancel ()

messagebox kullanımına ait örnek kod Şekil – 20’de ve kodun çıktısı da Şekil – 20.1’de gösterilmiştir.

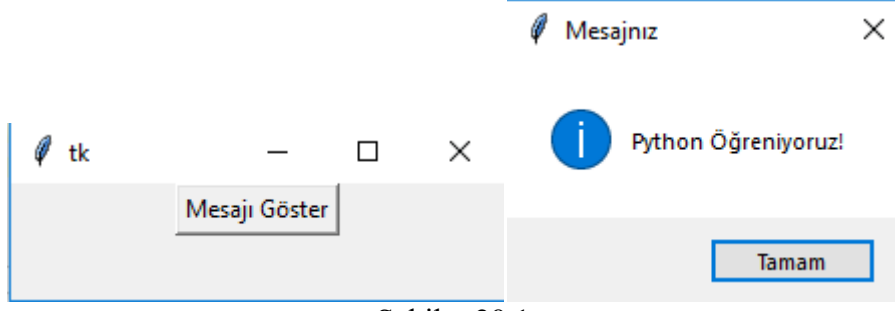
```
#messagebox Kullanımı
import tkinter as tk

win = tk.Tk()
def hello():
    tk.messagebox.showinfo("Mesajınız!", "Python Öğreniyoruz!")

B1 = tk.Button(win, text = "Mesajı Göster", command = hello)
B1.pack()

win.mainloop()
```

Şekil – 20



Şekil – 20.1

**Faydalı Adresler:**

[https://www.tutorialspoint.com/python/python\\_gui\\_programming.htm](https://www.tutorialspoint.com/python/python_gui_programming.htm)

[http://www.baskent.edu.tr/~tkaracay/etudio/ders/prg/gui/gui\\_ndx.html](http://www.baskent.edu.tr/~tkaracay/etudio/ders/prg/gui/gui_ndx.html)

<http://infohost.nmt.edu/tcc/help/pubs/tkinter/web/index.html>

<https://docs.python.org/3/library/tkinter.html>