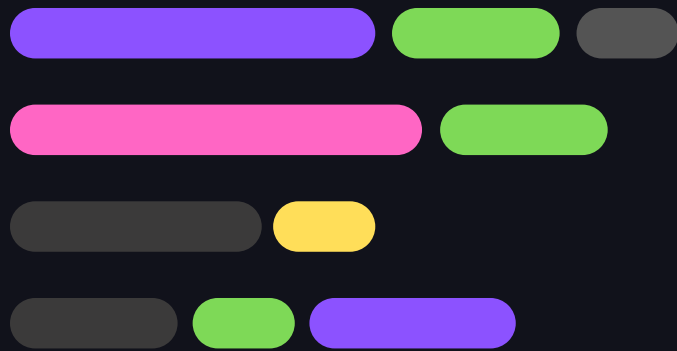
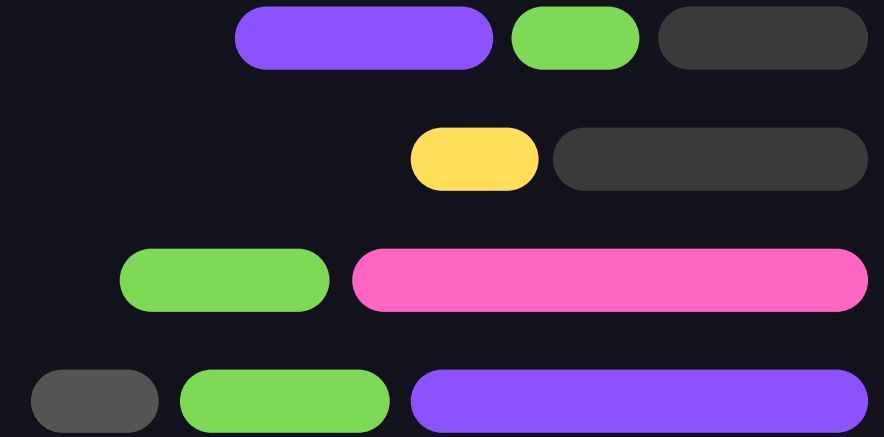




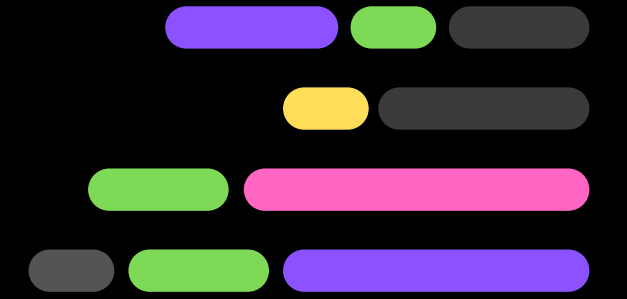
# { TEMEL PROGRAMLAMA II }

Samsun Üniversitesi  
Teknik Bilimler Meslek Yüksekokulu  
Arka-Yüz Yazılım Geliştirme Pr.





# Bu Haftanın Ders Kazanımları



01

String (Metin) İşlemleri

03

Proje Sunum Saati

02

Uygulama Saati

04

Proje Çalışma Saati

String (Metin)

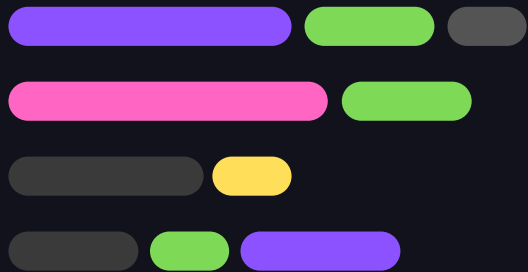
işlemleri



01

# String (Metin) İşlemleri

String nedir ?

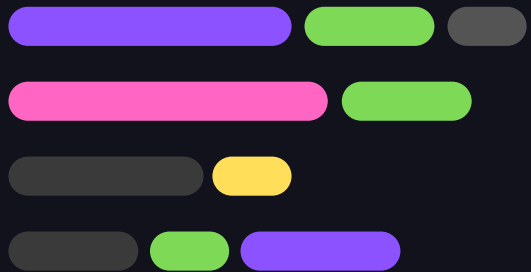




01

# String (Metin) İşlemleri

Python'da **stringler** karakterleri temsil eden baytları içeren listelerdir. Python doğrudan karakterleri temsil eden bir veri tipi içermez. Bu nedenle tek bir karakteri temsil eden veri, tek elemanlı bir string dizisidir. String verileri kolayca düzenlemek ve değiştirmek için dil içinde gelen gömülü fonksiyonlar vardır.



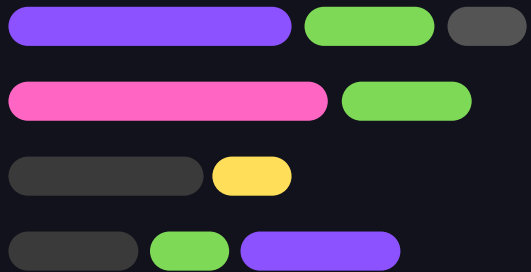


01

# String (Metin) İşlemleri

## String Verileri Birleştirme

Python da 2 string veri nasıl birleştirilir ?



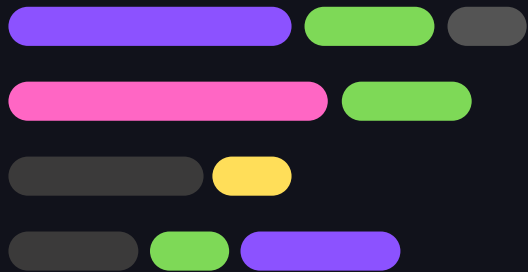


01

# String (Metin) İşlemleri

## String Verileri Birleştirme

```
ad = "Cüneyt"  
soyad = "Arkın"  
ad_soyad = ad + soyad  
print(ad_soyad)
```





01

# String (Metin) İşlemleri

## String Verileri Birleştirme

'Cüneyt Arkın' şeklinde çıktı verecek değişkeni tanımlayınız.

```
ad = "Cüneyt"  
soyad = "Arkın"  
  
print(ad_soyad)
```





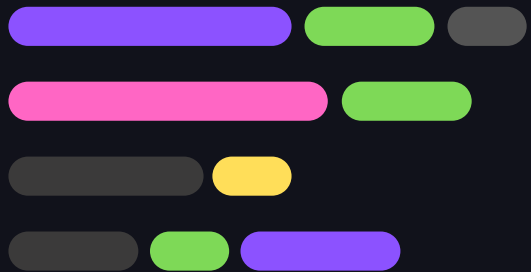
01

# String (Metin) İşlemleri

## String Verileri Birleştirme

'Cüneyt Arkın' şeklinde çıktı verecek değişkeni tanımlayınız.

```
ad = "Cüneyt"  
soyad = "Arkın"  
ad_soyad = ad + ' ' + soyad  
print(ad_soyad)
```





01

# String (Metin) İşlemleri

## String Verileri Birleştirme

Yandaki kodun çıktısı ne verir ?

```
yas = 20  
print("Yaşım: " + yas)
```





01

# String (Metin) İşlemleri

## String Verileri Birleştirme

String ve integer veriyi birleştirmek istersek integer veriyi string'e dönüştürmemiz gerekmektedir.

```
yas = 20
```

```
print("Yaşım: " + str(yas))
```





01

# String (Metin) İşlemleri

## String Verileri Birleştirme

Yandaki kodun çıktısı ne verir ?

```
yas = input("Yaş: ")  
print("Yaşınız: " + yas)
```





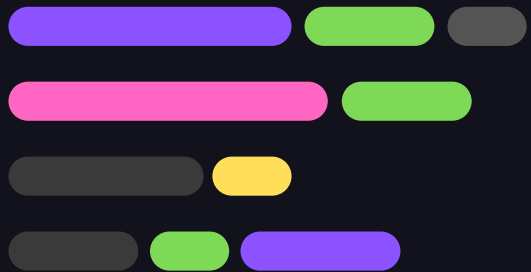
01

# String (Metin) İşlemleri

## String Verileri Birleştirme

Yandaki kodun çıktısı ne verir ?

```
text = "Python " * 3  
print(text)
```



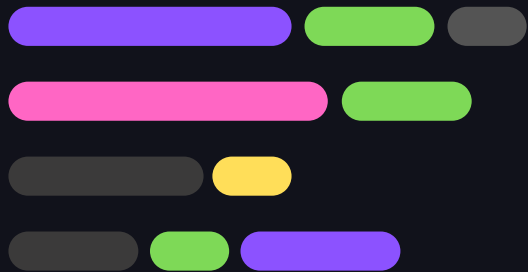


01

# String (Metin) İşlemleri

## String Veri İçindeki Bir Karaktere Erişme

Stringler, karakterlerden oluşmuş listeler olduğundan köşeli parantez ( `[]` ) ve index sayısı ile istenilen karaktere rahatça erişebilir.





01

# String (Metin) İşlemleri

## String Veri İçindeki Bir Karaktere Erişme

Yandaki kodun çıktısı ne verir ?

```
sehir = "Gaziantep"  
karakter = sehir[3]  
print(karakter)  
print(sehir[0])  
print(sehir[5])
```





01

# String (Metin) İşlemleri

## String Veri İçindeki Bir Karaktere Erişme

Yandaki kodun çıktısı ne verir ?

```
metin = "Python"  
print(metin[-1])
```





01

# String (Metin) İşlemleri

## String Veri İçindeki Bir Karaktere Erişme

Yandaki kodun çıktısı ne verir ?

```
metin = "Python"  
print(metin[10])
```





01

# String (Metin) İşlemleri

## String Veri İçindeki Bir Karaktere Erişme

Yandaki kodun çıktısı ne verir ?

```
metin = "Python"  
metin[0] = "A"
```



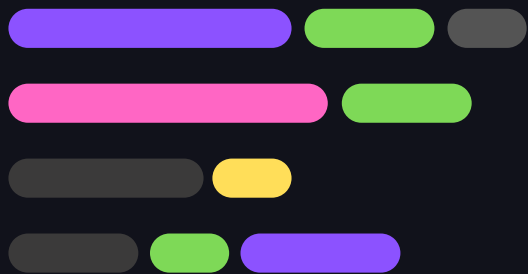


01

# String (Metin) İşlemleri

## String Verinin Uzunluğu

Listelere uzunluğu nasıl buluyoruz?





01

# String (Metin) İşlemleri

## String Verinin Uzunluğu

Yandaki kodun çıktısı ne verir ?

```
metin = "Python Ders"  
print(len(metin))
```





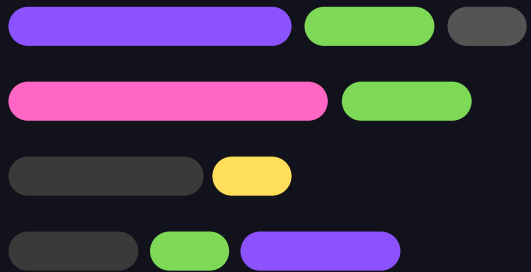
01

# String (Metin) İşlemleri

## String Verinin Uzunluğu

Yandaki kodu pc nizde çalıştırınız.

```
sehir = "Gaziantep"  
print("sehir değişken boyutu: ", len(sehir))  
boyut=len(sehir)  
son_karakter=sehir[boyut-1]  
print("Son karakter: ", son_karakter)  
for harf in sehir:  
    print(harf)
```





01

# String (Metin) İşlemleri

## String Verinin Uzunluğu

Yandaki kodu pc nizde çalıştırınız.

```
sehir deęişken boyutu: 9
Son karakter: p
G
a
z
i
a
n
t
e
p
```

```
sehir = "Gaziantep"
print("sehir deęişken boyutu: ", len(sehir))
boyut=len(sehir)
son_karakter=sehir[boyut-1]
print("Son karakter: ", son_karakter)
for harf in sehir:
    print(harf)
```



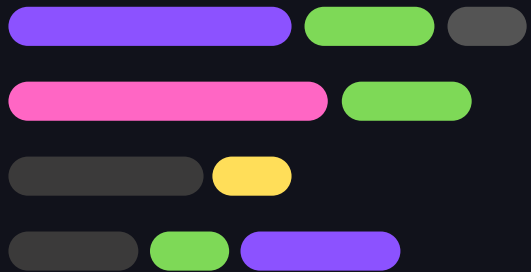


01

# String (Metin) İşlemleri

## String Veriyi Parçalama (Slice ) ve Bölme (Split)

slice() fonksiyonu ile köşeli parantezleri kullanarak metinden parçalar alınabilir. Bunun için köşeli parantez içine çekilmek istenilen karakter aralığının indisini girmek yeterlidir.



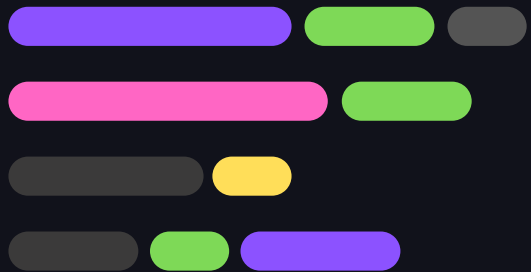


01

# String (Metin) İşlemleri

## String Veriyi Parçalama (Slice ) ve Bölme (Split)

`metin[başlangic:bitis]`  
başlangıç dahil, bitiş dahil DEĞİL





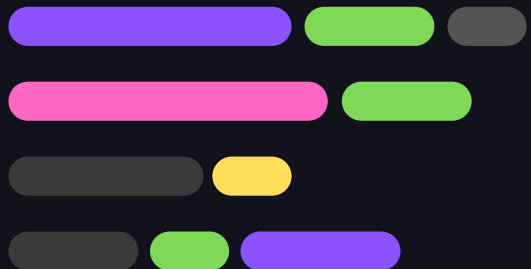
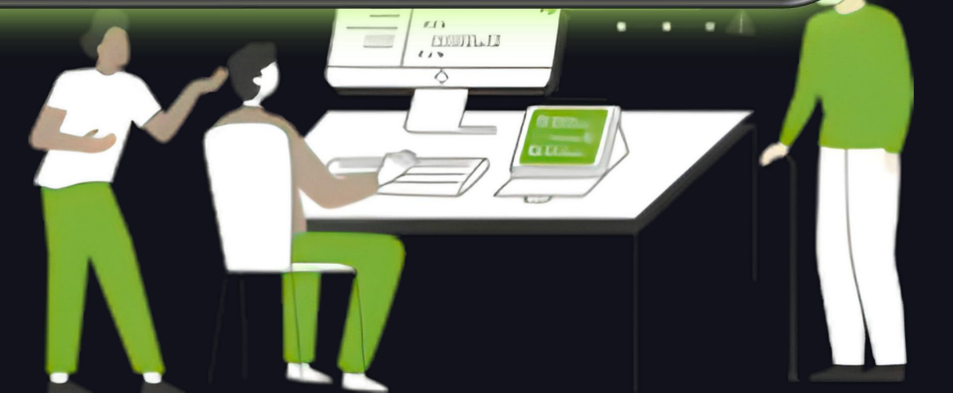
01

# String (Metin) İşlemleri

## String Veriyi Parçalama (Slice ) ve Bölme (Split)

Yandaki kodun çıktısı ne verir ?

```
metin = "Python"  
print(metin[0:3])
```





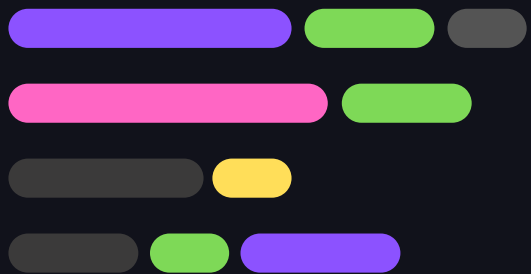
01

# String (Metin) İşlemleri

## String Veriyi Parçalama (Slice ) ve Bölme (Split)

Yandaki kodun bilgisayarınızda çalıştırınız.

```
veri = "Bilişim Teknolojileri"  
print(veri)  
print(veri[2:8])  
print(veri[2:])  
print(veri[5:15])  
print(veri[:12])
```





01

# String (Metin) İşlemleri

## String Veriyi Parçalama (Slice ) ve Bölme (Split)

Yandaki kodun bilgisayarınızda çalıştırınız.

```
veri = "Bilişim Teknolojileri"  
metin="ilkbahar,yaz,sonbahar,kış"  
veri_bolum = veri.split(" ")  
metin_bolum = metin.split(",")  
metin_bolum_ikieleman = metin.split(",",1)  
print(veri_bolum)  
print(metin_bolum)  
print(metin_bolum_ikieleman)
```





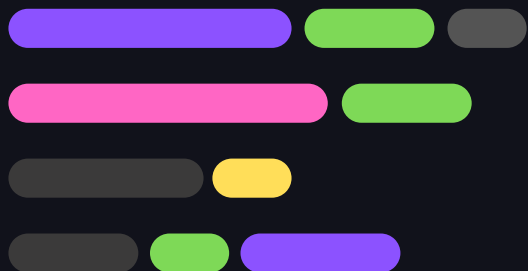
01

# String (Metin) İşlemleri

## String Veriyi Parçalama (Slice ) ve Bölme (Split)

Yandaki kodun bilgisayarınızda çalıştırınız.

```
liste = [1, 2, 3, 4, 5]
for i in liste:
    print(i)
```





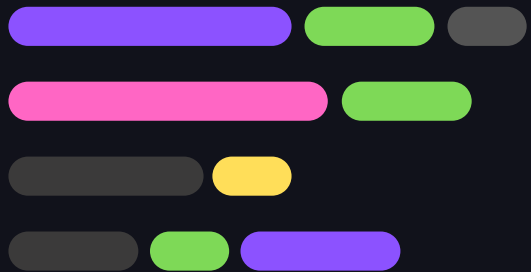
01

# String (Metin) İşlemleri

## String Veriyi Parçalama (Slice ) ve Bölme (Split)

Yandaki kodun bilgisayarınızda çalıştırınız.

```
liste = [1, 2, 3, 4, 5]
for i in liste:
    print(i, end=" ")
```





01

# String (Metin) İşlemleri

## String Veriyi Parçalama (Slice ) ve Bölme (Split)

print() fonksiyonunda kullanılan end parametresi, ekrana yazdırılan ifadenin sonunda ne olacağını belirler. Varsayılan olarak her print() satır sonuna (yeni satıra) geçer, ancak end=' ' gibi bir kullanım ile çıktının nasıl biteceğini değiştirebiliriz. Tırnak içinde yazılan ifade, çıktının sonuna eklenir ve bir sonraki yazdırma işlemi onun hemen ardından devam eder.





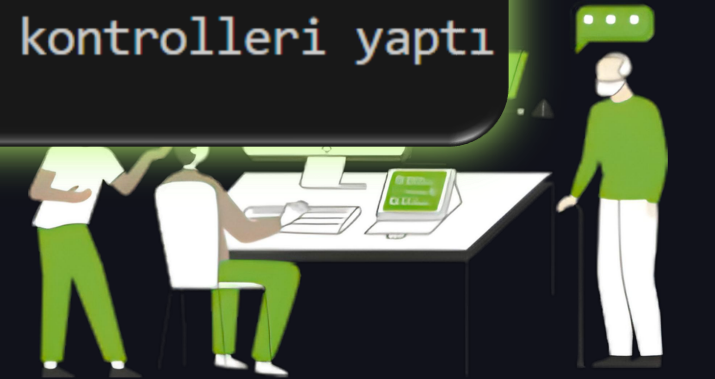
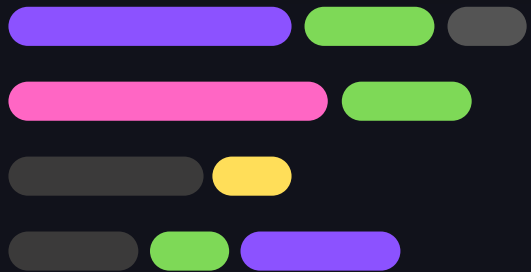
01

# String (Metin) İşlemleri

## String Veriyi Parçalama (Slice ) ve Bölme (Split)

Kullanıcıda bir metin alınız (internetten bulduğunuz bir metni, haberi yapıştırabilirsiniz), daha sonra sansürlemek istediğiniz kelimeyi alınız ve sansürlenmiş yeni metni ekrana yazdırınız.

```
Bir metin giriniz: Dün projeyi teslim etmek için Ali ile birlikte geç saate kadar çalıştık. Sabah olunca Ali son kontrolleri yaptı ve dosyayı gönderdi.  
Sansürlemek istediğiniz kelime: Ali  
Dün projeyi teslim etmek için ***** ile birlikte geç saate kadar çalıştık. Sabah olunca ***** son kontrolleri yaptı ve dosyayı gönderdi.
```



01

# String (Metin) İşlemleri

## String Veriyi Parçalama (Slice ) ve Bölme (Split)

Kullanıcıda bir metin alınız (internetten bulduğunuz bir metni, haberi yapıştırabilirsiniz), daha sonra sansürlemek istediğiniz kelimeyi alınız ve sansürlenmiş yeni metni ekrana yazdırınız.

```
metin = input("Bir metin giriniz: ")
sansur=input("Sansürlemek istediğiniz kelime: ")

kelimeler = metin.split(" ")

for kelime in kelimeler:
    if kelime == sansur:
        print("*****", end=" ")
    else:
        print(kelime, end=" ")
```



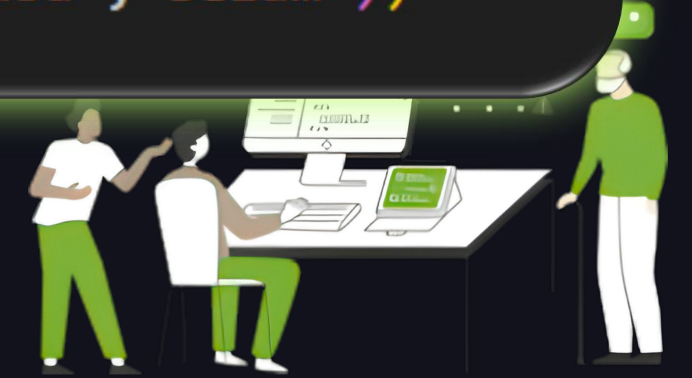
01

# String (Metin) İşlemleri

## String Veri İçinde Karakter Değiştirme, Karakter Ekleme ve Çıkarma

Yandaki kodun bilgisayarınızda çalıştırınız.

```
kelime = "Bilişim"  
cumle = "Merhaba Dünya"  
print(kelime)  
print(kelime.replace("i","o"))  
print(cumle)  
print(cumle.replace("Merhaba","Selam"))
```





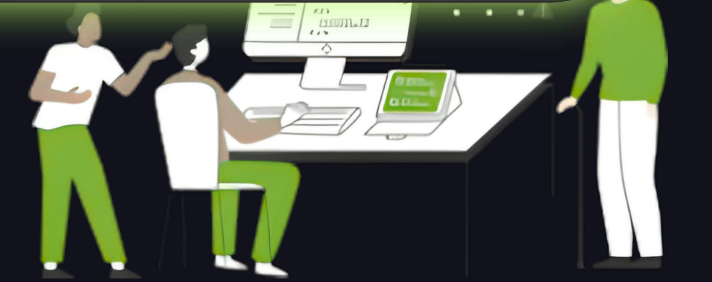
01

# String (Metin) İşlemleri

## String Veri İçinde Karakter Değiştirme, Karakter Ekleme ve Çıkarma

replace() Metin içindeki bir kısmı değiştirir.  
metin.replace(eski, yeni)

```
kelime = "Bilişim"  
cumle = "Merhaba Dünya"  
print(kelime)  
print(kelime.replace("i","o"))  
print(cumle)  
print(cumle.replace("Merhaba","Selam"))
```





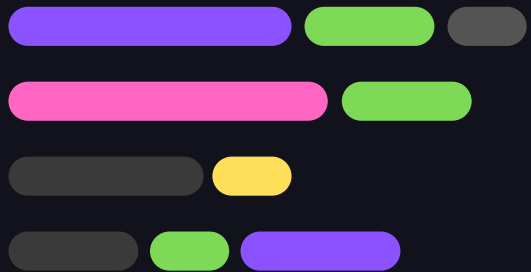
01

# String (Metin) İşlemleri

## String Veri İçinde Karakter Değiştirme, Karakter Ekleme ve Çıkarma

Yandaki kodun bilgisayarınızda çalıştırınız.

```
metin = "elma elma elma"  
print(metin.replace("elma", "armut", 1))
```





01

# String (Metin) İşlemleri

## String Veri İçinde Karakter Değiştirme, Karakter Ekleme ve Çıkarma

Kullanıcıda bir metin alınız (internetten bulduğunuz bir metni, haberi yapıştırabilirsiniz) daha sonra güncellemek istediği kelimeyi ve güncelleme kelimesini alınız. Ardından güncel metni ekrana yazdırınız.





01

# String (Metin) İşlemleri

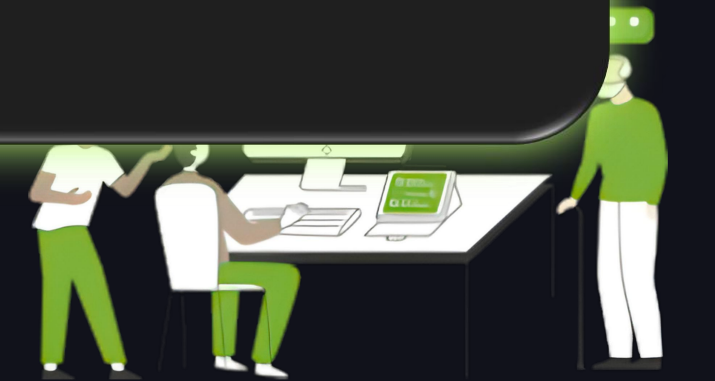
## String Veri İçinde Karakter Değiştirme, Karakter Ekleme ve Çıkarma

Kullanıcıda bir metin alınız (internetten bulduğunuz bir metni, haberi yapıştırabilirsiniz) daha sonra güncellemek istediği kelimeyi ve güncelleme kelimesini alınız. Ardından güncel metni ekrana yazdırınız.

```
metin = input("Bir metin giriniz: ")
eski_kelime = input("Güncellemek istediğiniz kelimeyi giriniz: ")
yeni_kelime = input("Yeni kelimeyi giriniz: ")

guncel_metin = metin.replace(eski_kelime, yeni_kelime)

print("Güncellenmiş metin:")
print(guncel_metin)
```





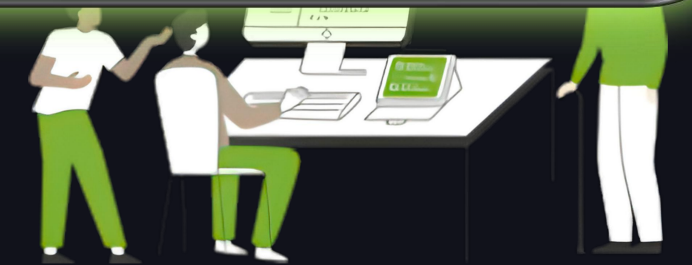
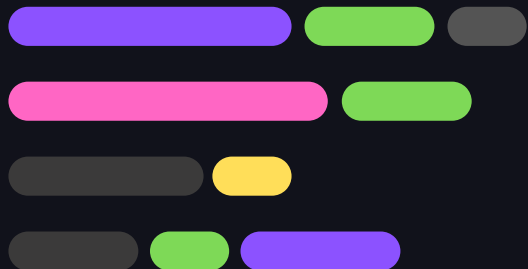
01

# String (Metin) İşlemleri

## String Veri İçinde Karakter Değiştirme, Karakter Ekleme ve Çıkarma

Yandaki kodun bilgisayarınızda çalıştırınız.

```
cumle = " bilişim teknolojilerine giriş "  
print(cumle.strip())  
print(cumle.strip(" bil"))  
print(cumle.strip("prg"))  
cumle2 = 'python çok kullanışlı'  
print(cumle2.strip("pyt"))
```





01

# String (Metin) İşlemleri

## String Veri İçinde Karakter Değiştirme, Karakter Ekleme ve Çıkarma

Yandaki kodun bilgisayarınızda çalıştırınız.

```
kelime = "Gaziantep"  
print(".").join(kelime))
```





01

# String (Metin) İşlemleri

**String Veri İçinde Bir Karakterin Yerini veya Metnin Karakteri İçerip İçermediğini Bulma**

Yandaki kodun bilgisayarınızda çalıştırınız.

```
kelime = "Bilişim Teknolojileri"  
print(kelime.find('m'))  
print(kelime.find('no'))  
print(kelime.find('i',4))  
print(kelime.find('i',1,7))  
print(kelime.find('z'))
```





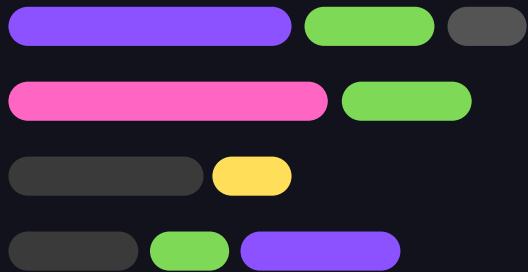
01

# String (Metin) İşlemleri

**String Veri İçinde Bir Karakterin Yerini veya Metnin Karakteri İçerip İçermediğini Bulma**

Yandaki kodun çıktısı ne verir ?

```
kelime = "Gaziantep"  
print("G" in kelime)  
print("g" in kelime)
```





01

# String (Metin) İşlemleri

## String Veri İle Büyük ve Küçük Harf Değişimi Yapma

Yandaki kodun bilgisayarınızda çalıştırınız.

```
kelime = 'Bilişim teknolojileri'  
print(kelime.upper())  
print(kelime.lower())  
print(kelime.capitalize())  
print(kelime.title())  
print(kelime.swapcase())
```



Uygulama Saati



02

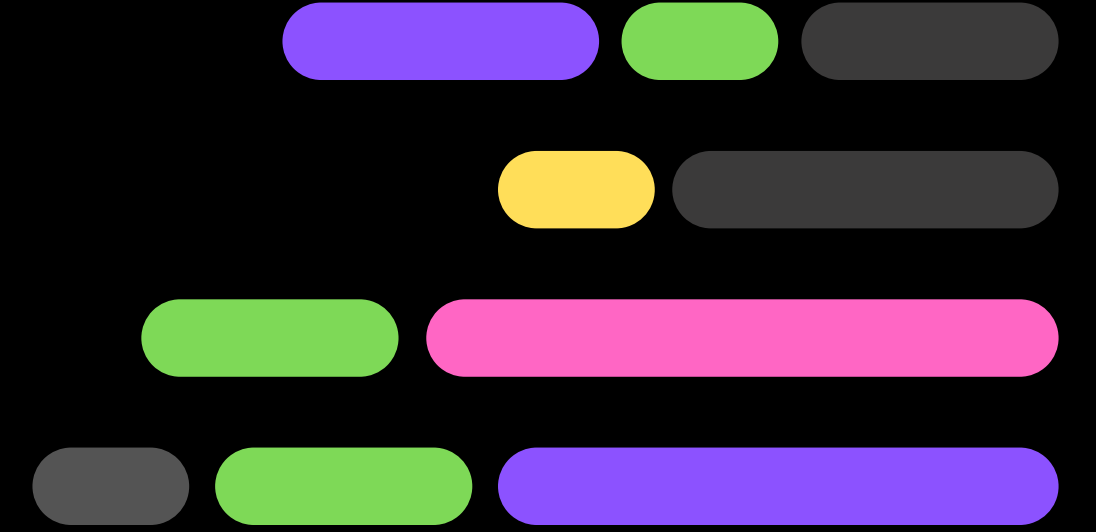
## Uygulama Saati

*Kullanıcıdan ad ve soyad alınacaktır.*

*Tüm harfler küçük yapılacaktır.*

*Araya nokta konularak  
birleştirilecektir.*

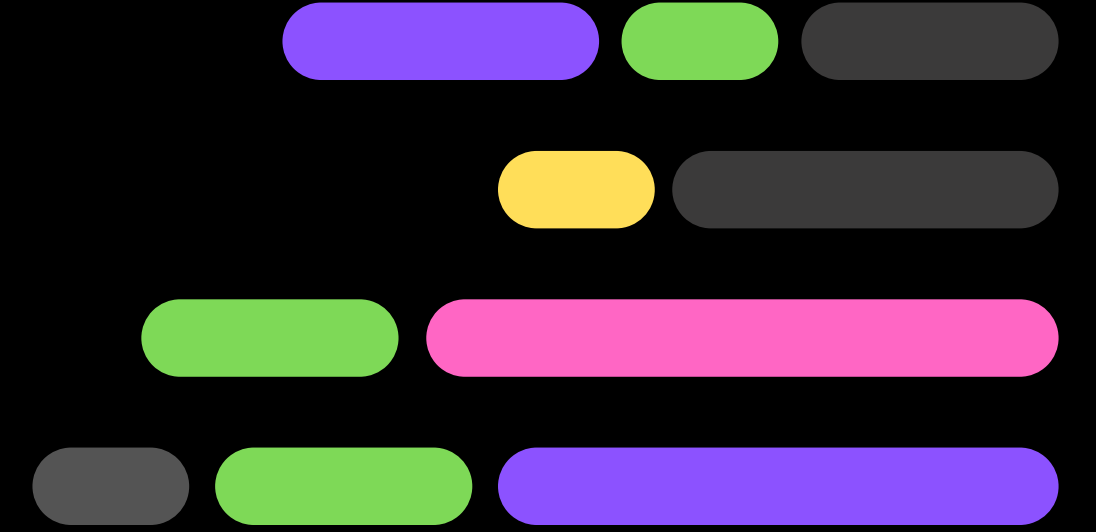
*Örnek çıktı: ahmet.yilmaz*





02

## Uygulama Saati



*Kullanıcıdan ad ve soyad alınacaktır.  
Tüm harfler küçük yapılacaktır.  
Araya nokta konularak birleştirilecektir.  
Örnek çıktı: ahmet.yilmaz*

```
ad = input("Ad: ")
soyad = input("Soyad: ")

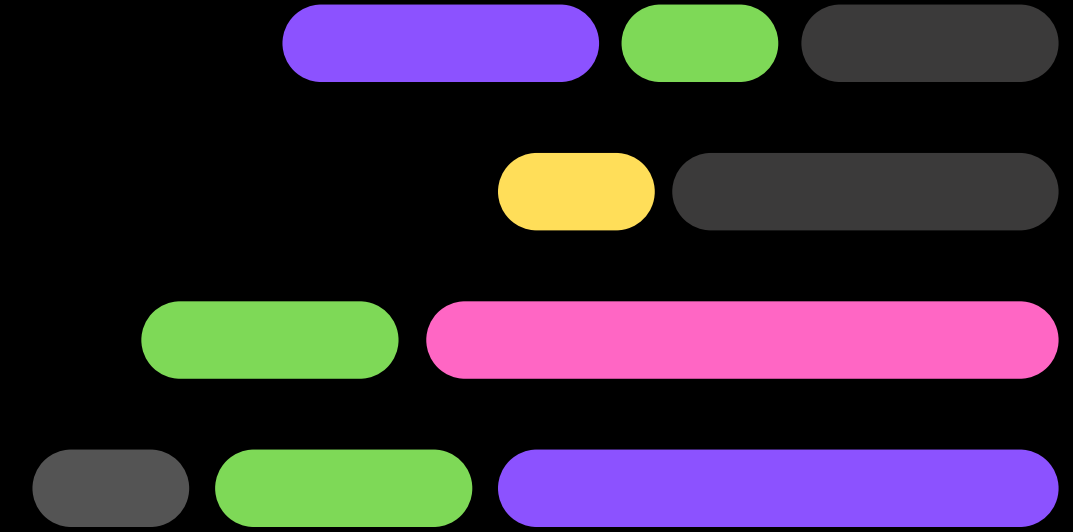
kullanici_adi = ad.lower() + "." + soyad.lower()

print("Kullanıcı adı:", kullanici_adi)
```



02

Uygulama Saati

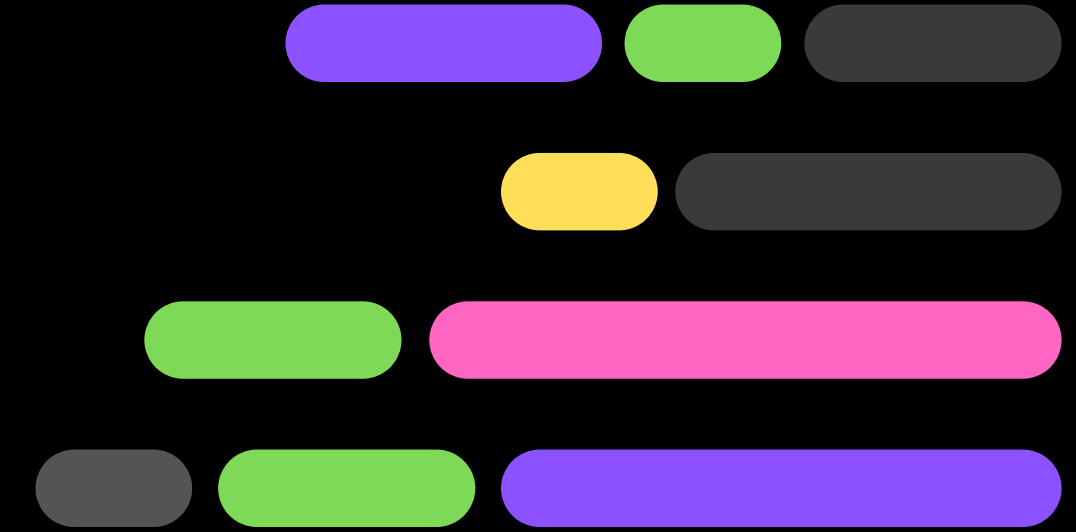


*Kullanıcıdan bir kelime alınacaktır.  
Kelimenin ortasındaki karakter  
yazdırılacaktır.*



02

## Uygulama Saati



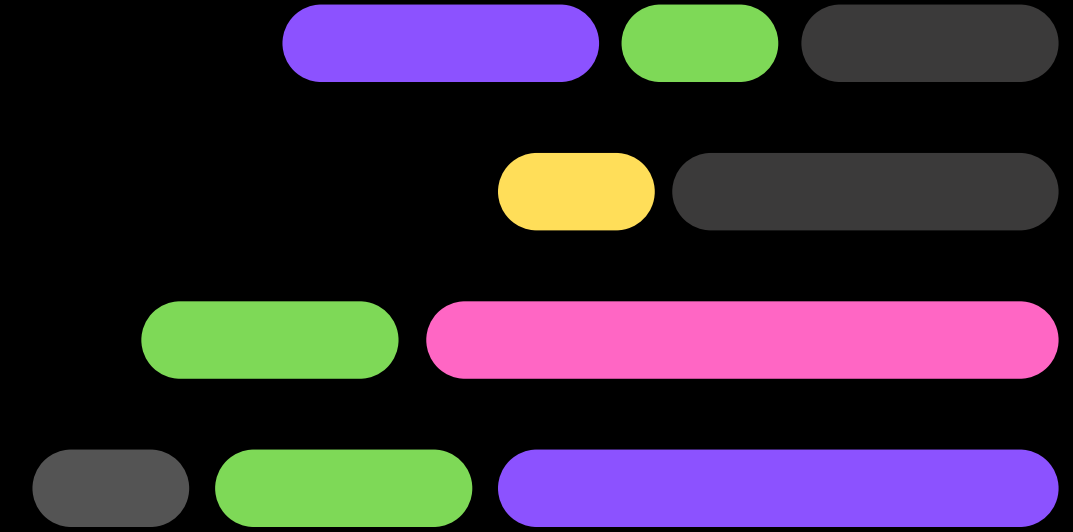
*Kullanıcıdan bir kelime alınacaktır.  
Kelimenin ortasındaki karakter  
yazdırılacaktır*

```
kelime = input("Kelime: ")  
orta_index = len(kelime) // 2  
print("Orta harf:", kelime[orta_index])
```



02

Uygulama Saati

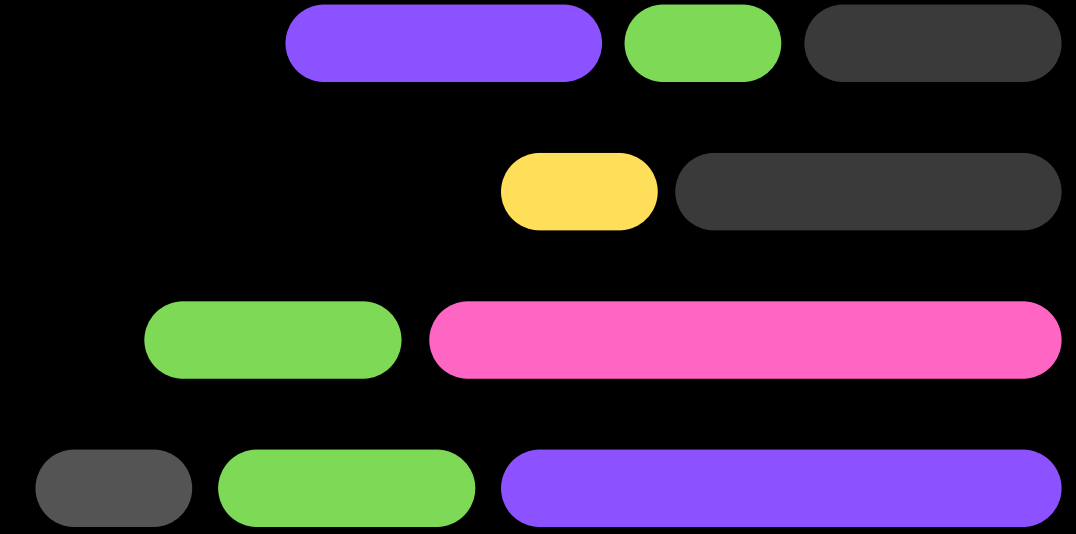


*Kullanıcıdan e-posta alınacaktır.  
İçinde "@" varsa geçerli sayılacaktır.*



02

## Uygulama Saati



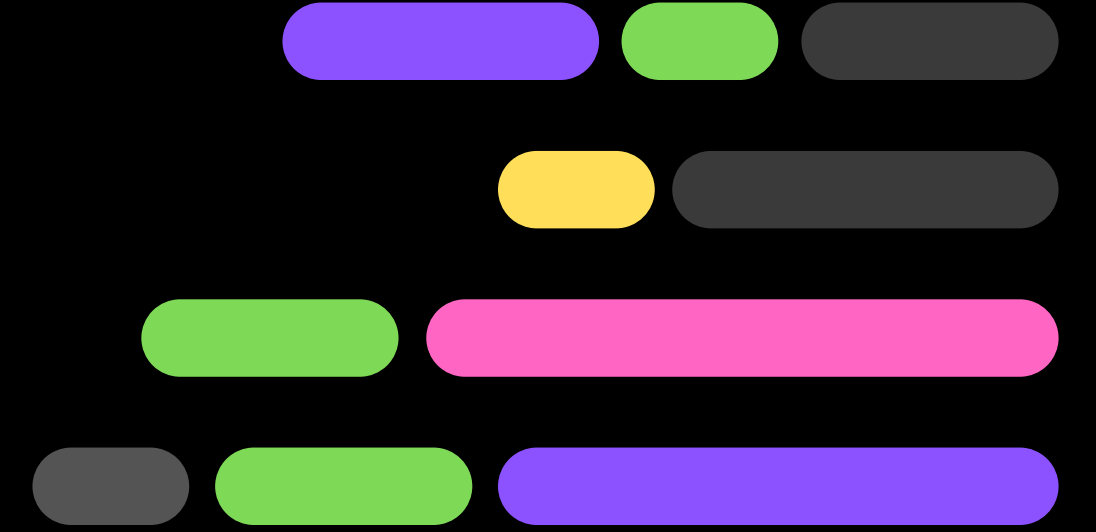
*Kullanıcıdan e-posta alınacaktır.  
İçinde "@" varsa geçerli sayılacaktır.*

```
mail = input("E-posta: ")  
if "@" in mail:  
    print("Geçerli e-posta")  
else:  
    print("Geçersiz e-posta")
```



02

Uygulama Saati



*Kullanıcıdan bir cümle alınacaktır.  
Kaç kelime olduğu hesaplanacaktır.*



02

## Uygulama Saati

*Kullanıcıdan bir cümle alınacaktır.  
Kaç kelime olduğu hesaplanacaktır.*

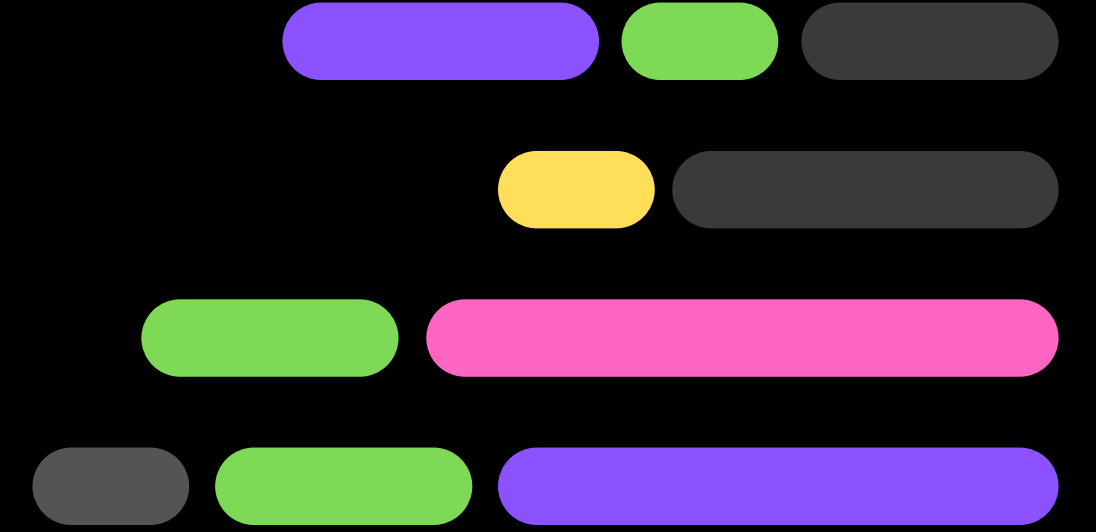
```
cumle = input("Cümle: ")  
kelimeler = cumle.split()  
print("Kelime sayısı:", len(kelimeler))
```



02

## Uygulama Saati

*Kullanıcıdan bir metin alınacaktır.  
Metindeki en uzun kelimeyi bulan kodu  
yazınız.*





02

## Uygulama Saati

*Kullanıcıdan bir metin alınacaktır.  
Metindeki en uzun kelimeyi bulan kodu yazınız.*

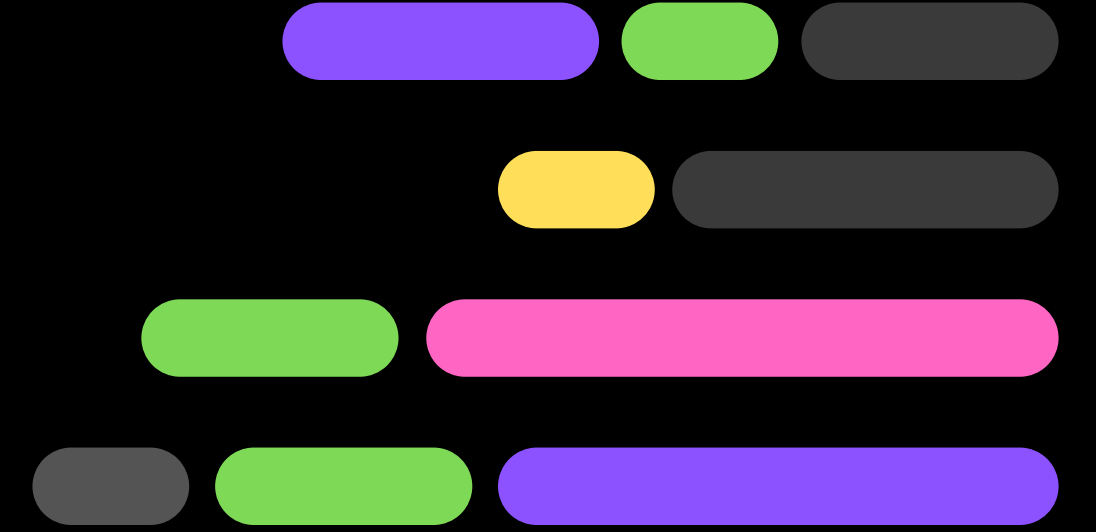
```
1  cumle = input("Bir cümle giriniz: ")
2
3  kelimeler = cumle.split()
4
5  en_uzun = ""
6
7  for kelime in kelimeler:
8      if len(kelime) > len(en_uzun):
9          en_uzun = kelime
10
11  print("En uzun kelime:", en_uzun)
12
```



02

## Uygulama Saati

*Kullanıcıdan sürekli olarak bir dosya adı alınız. Kullanıcı "q veya Q" girene kadar program çalışmaya devam etsin. Girilen dosya adında "?", "\*", "/" ve "\\" karakterlerinden herhangi biri bulunuyorsa bunun geçersiz olduğunu belirtiniz ve hangi karakterin hataya sebep olduğunu ekrana yazdırınız. Eğer bu karakterler yoksa dosya adının geçerli olduğunu yazdırınız.*





02

## Uygulama Saati

Kullanıcıdan sürekli olarak bir dosya adı alınız. Kullanıcı "q veya Q" girene kadar program çalışmaya devam etsin. Girilen dosya adında "?", "\*", "/" ve "\\" karakterlerinden herhangi biri bulunuyorsa bunun geçersiz olduğunu belirtiniz ve hangi karakterin hataya sebep olduğunu ekrana yazdırınız. Eğer bu karakterler yoksa dosya adının geçerli olduğunu yazdırınız.

```
yasakli = ["?", "*", "/", "\\"]  
while True:  
    dosya_adi = input("Dosya adı giriniz (q çıkış): ")  
  
    if dosya_adi.lower() == "q":  
        break  
  
    hatali = False  
  
    for karakter in dosya_adi:  
        if karakter in yasakli:  
            hatali = True  
            print("Geçersiz karakter içeriyor:", karakter)  
            break  
  
    if not hatali:  
        print("Dosya adı geçerli")
```



Son...