

STRING (Karakter Dizisi veya Metin) VERİLER ÜZERİNDE İŞLEMLER

String verideki karakterlerin konumlarına erişmek.

String verideki her karakterin bir adresi vardır adres 0 dan başlar.

ÖRNEK

```
metin="Python"
print(metin[0])  #0. karakter P karakteridir
print(metin[3])  #3. karakter h karakteridir
print(metin[5])  #5. karakter n karakteridir
```

```
P
h
n
>>> |
```

String verilerde negatif indexleme

String en son karakterin adresi [-1] sondan 2.karakterin [-2] şeklinde devam eder

ÖRNEK

```
metin="python"
print(metin[-1])  # -1 en son karakterdir >>"n" yazar
print(metin[-2])  # -2 sondan 2. karakterdir >> "o" yazar
```

String Verilerde Belirli bir dilimi almak

String verideki belli bir karakter dilimini almak için başlangıç ve bitiş adreslerini yazmalısınız. Bitiş adresindeki karakter alınmaz.

ÖRNEK

```
metin="Python"
print(metin[0:3])  #Pyt yazar 3. karakter h yi yazmaz
print(metin[2:5])  #tho yazar 5. karakter n yi yazmaz
print(metin[-1])   #son karakteri yazar n
print(metin[-3:-1]) #sondan -3 ten -1 e kadar olan karakterleri yazar >>>ho
```

```
metin="python"
print(metin[:4])
# baştan itibaren 4. karaktere kadar alır 4 karakter dahil değil
#>> pyth çıktısını verir
```

```
print(metin[3:])
#3. karakter dahil sonuna kadar tüm karakterleri yazar
#hon çıktısını verir
```

ÖRNEK:

“Darıca Fen Lisesi “ yazısının ilk harflerini alarak kısaltma şeklinde yazalım

```
okulAd="Darıca Fen Lisesi"
print(okulAd[0],okulAd[7],okulAd[11])    #sonuc DFL şeklinde yazacaktır
```

ÖRNEK:

Girilen bir yazıyı tersten yazmak için gerekli kodları yazalım

```
yazi=input("Yazı giriniz:")
print(yazi[::-1])

Yazı giriniz:Darıca Fen Lisesi
isesiL neF acıraD
>>> |
```

String verideki karakter sayısını öğrenmek

String bir verideki karakter sayısını öğrenmek için len() fonksiyonu kullanılır.

```
okulAd="Darıca Fen Lisesi"
karakterSayisi=len(okulAd)
print(karakterSayisi)    # 17 sonucunu verecektir boşluklar dahildir
```

String verilerin başındaki ve sonundaki boşlukları kaldırmak.

Bazı durumlarda string verilerin önündeki ve sonundaki boşlukları kaldırmak gerekebilir. Bu durumlarda strip() metodu kullanılmalıdır.

```
yazi="    Darıca Fen Lisesi    "
print(yazi)    # boşlukları alınmamış hali
print(yazi.strip())    #baştaki ve sondaki boşluklar alınmış hali
```

```
Darıca Fen Lisesi
Darıca Fen Lisesi
```

String verileri küçük harfe çevirmek ve lower () metodunun Kullanılması

```
metin="MERHABA"
print(metin.lower())    # çıktı merhaba şeklinde olacaktır
```

String verileri BÜYÜK harfe çevirmek ve upper() metodunun Kullanılması

```
metin="merhaba"
print(metin.upper())    #çıktı MERHABA şeklinde olacak
```

String verilerde küçük harfleri büyük büyük harfleri küçük yapmak. Swapcase() Kullanımı.

```
metin="Merhaba Python"  
print(metin.swapcase()) #çıktı mERHABA pYTHON şeklinde olacaktır
```

String verilerde kelimelerin ilk harflerini büyük yapmak ve title() metodunu kullanmak

String verilerdeki kelimelerin ilk harflerini başlıklarda olduğu gibi büyük harfe dönüştürmek için title() metodu kullanılır.

```
metin="string metodlarının kullanılması"  
print(metin.title())
```

Program çalıştırıldığında çıktısı aşağıdaki gibi olacaktır.

```
String Metodlarının Kullanılması
```

String verilerdeki belirtilen kelimenin kaç adet olduğunu bulmak count() metodunun kullanılması

```
metin="Pazardan elma aldım. En sevdiğim meyve elmadır."  
print(metin.count("elma")) #sonuç 2 çıkacaktır.Metindeki elma sayısı 2 dir
```

String verileri Hizalmak

```

metin="MERHABA"
print(metin.center(30)) #metni 30 karakterlik alan içerisinde ortalar
print(metin.center(30,"-")) #metni ortalar baş ve sonuna - ekler
print(metin.ljust(30,"-")) #metni sola hizalar
print(metin.rjust(30,"-")) #metni sağa hizalar

```

```

      MERHABA
-----MERHABA-----
MERHABA-----
-----MERHABA

```

String Verilerde Kelime veya Karakter değiştirme işlemlerinin yapılması. replace() metodunun kullanılması.

Kullanımı metin.replace(eskideger,yenideger,sayı)

```

metin="Pazardan kırmızı elma , yeşil elma ve sarı elma aldım"
print(metin) #değişim yapılmamış hali
print(metin.replace("elma","erik")) #tüm elmalar erik olarak değişir
print(metin.replace("elma","erik",2)) # ilk iki elma erik olacak

```

```

Pazardan kırmızı elma , yeşil elma ve sarı elma aldım
Pazardan kırmızı erik , yeşil erik ve sarı erik aldım
Pazardan kırmızı erik , yeşil erik ve sarı elma aldım

```

String verilerde format() metodunun kullanılması

Metin içerisindeki tutucuları { } kullanarak içerisine veri aktarımı yapılmaktadır.

```

ad="Ahmet"
soyad="USLU"
yas=27
metin="adım {} soyadım {} ve {} yaşındayım"
print(metin.format(ad,soyad,yas))

```

```

adım Ahmet soyadım USLU ve 27 yaşındayım

```

Aşağıdaki dilekçe hazırlama programını yazalım

```
File Edit Format Run Options Window Help
dilekce="""

                Darıca Fen Lisesi Müdürlüğüne
                        DARICA
        {} sınıfı {} nolu öğrencisiyim. Kent kart alabilmek için öğrenci belgesine
        ihtiyacım vardır.
        Gereğini bilgilerinize arz ederim

        Adı Soyadı      :{}
        Telefon         :{}
        Adres           :{}

"""

#Dilekçe şablonu üç tırnaklar içerisinde yazılarak değişkene atanmıştır

sinif=input("Sınıfınızı giriniz:")
no=input("Okul numaranızı giriniz:")
adSoyad=input("Adınız soyadınızı giriniz:")
telefon=input("Telefon Numaranızı giriniz:")
adres=input("Adresinizi Giriniz:")

print(dilekce.format(sinif,no,adSoyad,telefon,adres))

#format metodu ile string verilerdeki {} yer tutucularına
#veri gönderimi yapılabilmektedir
```

```
Sınıfınızı giriniz:9E
Okul numaranızı giriniz:989
Adınız soyadınızı giriniz:Tahsin GÜNEŞ
Telefon Numaranızı giriniz:53533322211
Adresinizi Giriniz:Darıca/Kocaeli

                Darıca Fen Lisesi Müdürlüğüne
                        DARICA
        9E sınıfı 989 nolu öğrencisiyim. Kent kart alabilmek için öğrenci belgesine
        ihtiyacım vardır.
        Gereğini bilgilerinize arz ederim

        Adı Soyadı      :Tahsin GÜNEŞ
        Telefon         :53533322211
        Adres           :Darıca/Kocaeli

>>> |
```