

YAPAY ZEKA



BİLİŞİM TEKNOLOJİLERİ TOPLULUĞU YAPAY ZEKA EĞİTİMİ DERS-2



SELÇUK
ÜNİVERSİTESİ

Diziler/Listeler(Array)



```
# boş liste değişkeni
```

```
liste = []
```

```
-----
```

```
# dizi/liste(array) oluşturmanın farklı bir tarzı
```

```
liste = list()
```

```
-----
```

```
# sadece int değerler tutan dizi/liste
```

```
liste = [1,2,3,4,5,6,7]
```

```
-----
```

```
# string tutan array
```

```
liste = list("İstanbul","Ankara","İzmir")
```



```
# farklı değişken türleri tutan liste isimli array değişkeni  
liste = [10,11,12,"Python",3.14,51.035]
```

```
-----  
# string parçalayarak dizi/liste oluşturma  
isim = 'YapayZeka'  
liste = list(isim)
```

```
-----  
# array indexleme /basit dizi  
liste = ["Elma","Armut","Kivi","Muz"]  
liste[2]
```



array indexleme / iç içe dizi

liste1 = [1,2,3]

liste2 = [4,5,6]

liste3 = [7,8,9]

yeniliste = [liste1,liste2,liste3]

yeniliste[1,2]



Sona tek bir eleman eklemek için **append**, başka bir listeyi eklemek için **extend**, belirli bir index'e yerleştirmek için **insert** fonksiyonları kullanılır.

`liste.append(eleman)`

`liste.extend(sequence)`

`liste.insert(index,eleman)`



-list tipi **reverse** metoduyla yerinde ters çevirilebilir
liste.reverse()

Ters çevirmek oldukça hızlıdır. Dolayısıyla listenin başına ekleme çıkarmalar yapacaksanız önce ters çevirip, sonra işlemleri yapıp, tekrar ters çevirerek zaman kazanabilirsiniz, çünkü listenin sonuna ekleme yapmak, başına ekleme yapmaktan çok daha hızlıdır.

liste.index(eleman) // parametre olarak verdiğimiz elemanın indexini verecektir

minik = min(liste)
buyuk = max(liste)

Pop() Fonksiyonu



```
dizi = [10, 20, 30, 40]  
dizi.pop(0)  
print(dizi)
```

Örnek Soru



→Kullanıcıdan 1-12 arasında bir değer alarak aldığı değer karşılığı olarak ay adını ekrana yazdırdıktan sonra yazılan ay adını diziden sildikten sonra dizinin tamamını ekrana yazdıran algoritmayı tasarlayın ve Python dili ile yazın.

Örnek Soru Çözüm:

```
aylar = ["Ocak", "Subat", "Mart", "Nisan", "Mayıs", "Haziran", "Temmuz", "Ağustos", "Eylül", "Ekim", "Kasım", "Aralık"]
for i in range(len(aylar)):
    print(i, "-->", aylar[i])
deger = int(input("Bir Sayı Giriniz"))
print(aylar[deger])
aylar.pop(deger)
print(aylar)
```

Numpy Kütüphanesi

Nedir?

- NumPy, Python' da bilimsel hesaplamalarda kullanılan bir python kütüphanesidir. NumPy genel olarak dizi/vektör/matris hesaplamaları için özelleşmiş bir kütüphane olup, büyük çaptaki veri kümeleri üzerinden işlem yapmayı kolaylaştırır. Genelde matematiksel işlemler yapılır ve tipik olarak bu tür operasyonlar, Python'un dahili dizilerini kullanarak mümkün olana oranla daha verimli ve daha az kodla yürütülür. NumPy kullanılarak istatistik işlemleri ve simülasyonlarda yapılabilir. NumPy paketinin temelinde, ndarray nesnesi vardır. Bu, homojen veri türlerinin n-boyutlu dizilerini kapsar, birçok işlem performans için derlenmiş kodda gerçekleştirilir. NumPy dizileri ile standart Python dizileri arasında birkaç önemli farklılık vardır. Bunlar:

Numpy Diziler(Arrays)



```
#importing  
import numpy as np
```

```
dizi1 = np.array([0,1,2,3,4,5,6,7]) #tek boyutlu dizi  
dizi2 = np.array([[0,1,2],[3,4,5]]) #iki boyutlu dizi  
dizi3 = np.empty([3,2], dtype = np.int) #integer tipte 3x2lik bos bir matris olusturduk. Bu dizideki 0'lar baslatılmadıkları için rastgele değerler gösterir.  
dizi4 = np.zeros((2,2), dtype = np.int) #2x2'lik 0'lardan oluşan bir matris.  
dizi5 = np.ones(5, dtype = np.int) #5 elemanlı 1'lerden oluşan bir dizi.  
dizi6 = np.random.rand(3, 4) #3x4 lük random array
```

Matris Dönüştürme :

```
dizi = np.array([0, 1, 2, 3, 4, 5, 6, 7]) # tek boyutlu dizi  
  
dizi = dizi.reshape(2, 4) # 2x3'lük matrise dönüştürüldü.  
  
dizi = dizi.reshape(8) # tekrar tek boyutlu dizi haline getirildi.
```

Otomatik Dizi Oluşturma:

```
dizi = np.arange(15) # 0 dan başlayıp 15'e kadar giden bir dizi oluşturur. 15 dahil değil.  
  
dizi = np.arange(4, 20) # 4 dan başlayıp 20'ye kadar giden bir dizi oluşturur. 20 dahil değil.  
  
dizi = np.arange(0, 50, 5) # 0 dan başlayıp 50'ye kadar 5'er 5'er giden bir dizi oluşturur. 50 dahil değil.
```

Dizinin Elemanlarında Dolaşma:

```
dizi = np.arange(10)

eleman5 = dizi[4]

dizi[4] = 15 # dizi elemanlarını değiştirebiliriz.

dizi1 = dizi[2:8] # 2. indeksten 8. indekse kadar olan sayılar (8 dahil değil).

dizi2 = dizi[0:8:2] # 0. indeksten 8. indekse kadar ikiser artarak.

dizi3 = dizi[2:] # 2. indeksten itibaren gerisini alır.

ikiboyutludizi = np.array([[1, 2, 3], [3, 4, 5], [4, 5, 6]])

dizi4 = dizi[1:] # iki boyutlu dizinin ilk satırını almaz. Çıktı [[3,4,5],[4,5,6]] olacaktır.
```

Dizilerde İşlemler:

```
a = np.arange(9, dtype=np.int).reshape(3, 3)

b = np.array([10, 10, 10])

print(np.add(a, b)) # dizilerin toplanması

print(np.subtract(a, b)) # çıkarılması

print(np.multiply(a, b)) # çarpılması

print(np.divide(a, b)) # bölünmesi
```

Dizilerde İşlemler:

```
a = np.array([1, 2, 3, 4])
b = np.array([10, 20, 30, 40])
c = a * b
print(c)
c = a + b
print(c)
c = a - b
print(c)
c = b / a
print(c)

a = np.array([[0.0, 0.0, 0.0], [10.0, 10.0, 10.0], [20.0, 20.0, 20.0], [30.0, 30.0, 30.0]])
b = np.array([1.0, 2.0, 3.0])

c = a + b
print(c)
```

Dizilerde Sıralama:

```
dizi = np.array([5, 7, 3, 1, 9, 8])

print(np.sort(dizi))

dizi2 = np.array([[9, 6], [9, 8]])
print(np.sort(dizi2, axis=0))
print(np.sort(dizi2, axis=1))
```


Soru:
İki dizi arasındaki fark nedir?

```
dizi1 = np.array([0, 1, 2, 3, 4, 5, 6, 7, 8, 9])  
print(dizi1.shape)  
print(dizi1.ndim)  
  
dizi2 = np.array([[0, 1, 2, 3, 4, 5, 6, 7, 8, 9]])  
print(dizi2.ndim)  
print(dizi2.shape)
```

Bazı İstatistiksel Bilgiler



Elimizde ki dizi $\rightarrow 13, 18, 13, 14, 13, 16, 14, 21, 13$

Mean = Ortalama

Median = Dizi sıralandıktan sonra sol ve sağdaki sayılar birer birer ortada bir ve iki sayı kalana kadar elenir. Ortada kalan bir tane sayı varsa median o sayıya eşittir. İki tane sayı varsa median bu iki sayının ortalamasına eşittir.

VARYANS \bar{X} :

Elimizde ki dizi $\rightarrow 3,4,8,9$

Mean = 6

Varyans \rightarrow İlk adımda dizideki bütün sayılardan ortalama çıkartılır.

$$3 - 6 = -3$$

$$4 - 6 = -2$$

$$8 - 6 = 2$$

$$9 - 6 = 3$$

\rightarrow Elde edilen tüm sayıların hepsinin karesi alınır

$$(-3)^2 = 9$$

$$(-2)^2 = 4$$

$$(2)^2 = 4$$

$$(3)^2 = 9$$

\rightarrow Tüm elde ettiğimiz sayılar toplanır

$$9 + 4 + 4 + 9 = 26$$

\rightarrow Karelerin toplamı elimizdeki sayı adedinin bir eksiğine bölünür

$$26 / (4-1) = 8,6$$

Varyans (sembolü \bar{X}) = 8,6

Standart Sapma (sembolü σ) = $\sqrt{\bar{X}} = \sqrt{8,6} = 2,93$

Belirttiğimiz İşlemlerin Numpy İle Çözümleri:

```
numpy_array = np.array([13, 18, 13, 14, 13, 16, 14, 21, 13])
#Mean
print(np.mean(numpy_array))
#Median
print(np.median(numpy_array))
#Varvans
print(np.var(numpy_array))
#Standart Sapma
print(np.std(numpy_array))
```