

```

"""None"""
sehir = None #BİR DEĞER VAR AMA BİZ O DEĞERİ BİLMİYORUZ
print("Değer", sehir, "Tip", type(sehir))

sehir = "istanbul"
print("Değer", sehir, "Tip", type(sehir))

"""Numeric(sayısal)"""
#integer
#float
#complex
#bool

say1 = 5
print(type(say1))
say1 = 2.5
print(type(say1))
say1 = 2+7j
print(type(say1))
say1 = True
print(type(say1))

say11 = 7.5
say12 = int(say11)
print(say11, type(say11))
print(say12, type(say12))

say13 = 10
say14 = float(say13)
print(say14, type(say14))

x = int(10)
y = float(15.0)
z = complex(2, 2)
t = bool(True)
print(x, y, z, t)

print(3<4)
anahtar = 3<4
print(anahtar, type(anahtar))

print(int(True))
print(int(False))
print(float(True))
print(float(False))

print(bool(10))
print(bool(1.2))
print(bool(-9))
print(bool(-8.9))

"""DİZİLER"""
#List
#Mutable(değiştirilebilir)
#Elemanların sıra numarası(sequence number) vardır
#[]

liste = [2, 6, 7, 9]
print(liste, type(liste))
print(*liste)

```

```

print(liste[3])
print(liste[3]*9)

liste = []
print(liste)
liste = list()
print(liste)
liste = list([1, 2, 3])
print(liste)

liste2 = [2, 3, 4]
print(liste2)
liste2[1]=15
print(liste2)

sayılar =[13, 16, 47, 79, 87, 99]
print(sayılar)
print(sayılar[0])
print(sayılar[4])
print(sayılar[-1])
print(len(sayılar))
print(sayılar[len(sayılar)-1])

isim = "berk batuhan"
isim = list(isim)
print(isim)
isimler = ["berk", "batuhan", "devran", "adı lazım değil"]
print(isimler)
print(isimler[-1])
print(isimler[len(isimler)-1])

karışıktpıçerenliste = [33, 45, "batuhan", True, None, {2,5}, {"595" :
"devran"}]
print(karışıktpıçerenliste)
print(karışıktpıçerenliste[-1])
print(type(karışıktpıçerenliste[-1]))
print(karışıktpıçerenliste[-1]["595"])
bilgiler = [isimler, sayılar]
print(bilgiler)
bilgiler = isimler+sayılar
print(bilgiler)
print(4*isimler)

isimler[2] = "fenerbahçe"
print(isimler)

sayılar[:2] = []
print(sayılar)
sayılar.append(123)
print(sayılar)
sayılar.insert(2,19)
print(sayılar)

sayılar.pop() # son elemanı siler
print(sayılar)
print(sayılar.pop()) #silinen elemanı yazdırır
x=sayılar.pop()
print(x)
print(sayılar.pop(2))
print(sayılar)
del sayılar[:2]

```

```

print(sayılar)
# del sayılar      sayıların tamamını siler

sayılar.extend([5, 12, 14])
print(sayılar)
sayılar.extend({5, 12, 14})
print(sayılar)
sayılar.extend("batuhan")
print(sayılar)
sayılar.extend({"batuhan": "595"})
print(sayılar)

sayılar = [1, 6, 0, 19, 67, 99]
print(min(sayılar))
print(max(sayılar))
print(sum(sayılar)) #toplamı verir
sayılar.sort(reverse=True)
print(sayılar)
print(sayılar[::-1])
sayılar.reverse()
print(sayılar)

#Tuple
#Immutable (değiştirilemez)
#Elemanların sıra numarası (sequence number) vardır
#()

tup = (4, 24, -67, 90)
print(tup, type(tup))
tup2 = ()
print(tup2)
tup3 = tuple((4, 6))
print(tup3)
print(tup[2])

tup = (14, 99, 67, 14)
print(tup)
print(type(tup))
print(tup[1])
print(tup[::1])
print(tup[::2])
print(tup[1::2])
print(tup.count(14))
print(tup.index(14))

tup9 = (1, 3, 5)
tup10 = (0, 7, "batuhan")
tup11 = tup9 + tup10
print(tup11)

#Set
#Mutable (değiştirilebilir)
#Elemanlarının sıra numarası (sequence number) yoktur
#İçerisinde aynı değerde elemanlar barındırmaz

bosSet = set()
print(type(bosSet))
bosSet2 = {}
print('set değil dict olur;', type(bosSet2))

```

```

set1 = {3, 5, 7, 9}
print(set1)
set1.add(111)
print(set1)
set1.remove(3)
print(set1)

set1 = {5, 7, 0, 9}
print(set1)
set1.remove(5)
print(set1)
set1.update({13, 18, 10})
print(set1)
set1.add(999)
print(set1)
harfler = set("batuhan")
print(harfler)
set2 = {9999, 66, 77, 0}
unionset = set2.union(set1)
print(unionset)

#String(karakter dizileri)
#immutable(değiştirilemez)
#Elemanların sıra numarası(sequence number) vardır.
# "", ''

karakter = "F"
print(karakter, type(karakter))
isim = "Batuhan"
print(isim[4])
"""isim[4] = "k"
print("isim")
"""# değiştirilemez

#Range
print(range(10))
print(list(range(0,10)))
print(list(range(1,10,2)))
print(*range(0,5))

#Dictionary(sözlük)
#mutable(değiştirilebilir)
#Elemanların sequence numberi yerine "KEY"leri vardır
#{'', ''}

bosSözlük = dict()
print("BOSSÖZLÜK;", bosSözlük, type(bosSözlük))
bosSözlük2 = {}
print("BOSSÖZLÜK2;", bosSözlük2, type(bosSözlük2))

sözlük = {'12211212121' : 'Berk Batuhan Devran', '43434343434' : 'Adı lazım değil'}
print(sözlük)
print(sözlük['12211212121'])
sözlük['12211212121'] = 'Batuhan'
print(sözlük)
print(*sözlük)
print(sözlük.keys())

```

```

print(sözlük.values())
print(sözlük.get('12211212121'))

sozluk = {1: "batuhan", 2: "berk", 3: "devran"}
print(sozluk)
print(sozluk[1])
print(sozluk.get(10)) #none
print(sozluk.get(10, "eleman bulunamadı"))
print(sozluk.keys())
print(sozluk.values())
print(sozluk.items())

emptyDict = {}
emptyDict2= dict()
anahtarlar= {"batuhan", "berk", "devran"}
print(anahtarlar)
değerler = ["Elektrik MÜH", "KONROL MÜH", "HABERLEŞME MÜH"]
bilgiler = dict(zip(anahtarlar, değerler))
print(bilgiler)
bilgiler["berrak"] = "yazılım müh"
bilgiler["berk"] = "kontrol ve otomasyon müh"
print(bilgiler)
del bilgiler["berrak"]
dersler = {"berk": "müh mat", "batuhan": ["matlab", "phyton"],
"devran": ["elmak", "plc", "scada"]}
print(dersler)
print(dersler["devran"][2])

sözlük1 = {123:"berk"}
sözlük2 = {1234:"batuhan"}#key 123 olsaydı berk yerine batuhanı islerdi
sözlük1.update(sözlük2)
print(sözlük1)
sözlük3 = sözlük1
print(sözlük1)
sözlük4 = {**sözlük3}
print(sözlük4)

### MATH MODÜLÜ

import math
print(math.sqrt(25))
print(math.pow(2,5))
print(math.pi)
print(math.ceil(2.3))
print(math.floor(2.3))
print(math.sin(math.pi/2))
print(math.cos(math.pi))

from math import * # * yerine istenen fonksiyon eklenebilir
print(tan(pi/2))
print(e)
e = 9
print(e)

import math as m
print(m.cos(0))
print(m.cos(pi))
print(m.cos(pi/2))

```

