

## BİLGİSAYAR BİLİMİ DERSİ PYTHON PROGRAMLAMA DİLİ ÇALIŞMA KÂĞIDI

- **Ekrana Metin Yazdırmak**

Ekranda metin yazdırmak istendiğinde print komutu kullanılır. Kullanımı aşağıda verilmiştir. Parantez içinde çift ya da tek tırnak kullanılabilir. Ayrıca Türkçe'ye özel harflerin(ı,ş,ç,ö,ğ,ü,ö) parantez içinde kullanılmasında bir sakınca yoktur.

```
print ("Deneme")
```

```
print ('Deneme')
```

- **Değişkenler ve Değerinin Yazdırılması**

Değişkenler programlama aşamasında hafızada bilgi tutmaya yarayan araçlardır. Değişkenlere bir isim verilerek gerektiğinde o isimlerle çağırılır. Kullanım şekli aşağıdaki gibidir.

```
a=10          #(bu örnekte a isimli değişkene 10 değeri verilmiştir.)
```

```
print (a)     #(değişkenin değerini yazdırmak istediğimizde tırnak kullanılmaz. Bu iki satır kod çalıştığında  
# ekranda 10 görüntülenir.
```

- **Değişken Türleri**

Değişkenler tuttukları verinin içeriğine göre türlere ayrılırlar. Aşağıda örnekleri verilmiştir.

```
a=10          #(a değişkeni tam sayı bir değer tuttuğu için integer türünde bir değişkendir.)
```

```
b=3.14        #(b değişkeni ondalıklı sayı bir değer tuttuğu için float türünde bir değişkendir.)
```

```
c="KAL-1977"   #(c değişkeninin değeri tırnak içine yazıldığı için string türünde bir değişkendir.)
```

```
d=True        #(d değişkeni sadece iki değer alabilen (True ya da False) boolean türünde bir değişkendir.)
```

Konudan farklı olarak; # işareti kodlama aşamasında koda etki etmeden açıklama yapılması istendiğinde kullanılır.

- **Python' da Dört İşlem**

Dört işlem yaparken +, -, \*, / sembolleri kullanılır. Burada çarpma sembolünün Matematikteki sembolünden farklı olduğunu ve bölme sembolünün yönünün bu şekilde olduğunu unutmamamız gerekir. Karışıklığa mahal vermemek için klavyenin en sağdaki kısmı kullanabiliriz. Kullanım örneği aşağıda verilmiştir.

```
x=20+10
```

```
y=20-10
```

```
z=20*10
```

```
w=20/10
```

```
print (x,y,z,w)    #Kodu çalıştırdığımızda ekranda 30,10,200,2 sonucu çıkacaktır.
```

- **Python'da Üs Alma ve Bölümden Kalanı Bulma**

```
a=3**20          # 3'ün 20.kuvveti bu şekilde alınır.
```

```
b=81%4           # 81'in 4'e bölümünden kalan bu şekilde alınır.
```

- **Python’da String Değişkenlere Özel İşlemler**

- **Bir Metnin İstenilen Sıradaki Karakterini Bulmak**

a="Çanakkale"[0] #1.sıradaki karakteri verir.

print (a) # a değişkeninin değeri "Ç" dir. Python dili karakterleri 0’dan başlayarak sayar.

a="Çanakkale"[0:5] #1-5 arası karakterler olan "Çanak" kelimesini alır.

print (a)

- **Bir Metnin Karakter Uzunluğunu Bulmak**

a="Çanakkale-1915"

print (len(a)) # ekranda sonuç olarak karakter sayısı olan 14 çıkacaktır. Burada dikkat etmemiz gereken #boşluk, özel işaret ve rakamların da sayılacağıdır.

- **Bir Metni Büyük ya da Küçük Harfe Çevirmek**

a="Python’u Çok Seviyorum"

print (a.upper()) #büyük hale çevirip ekranda gösterir.

print (a.lower()) #küçük hale çevirip ekranda gösterir.

- **Diğer Değişken Türlerini String Türüne Çevirme**

Stringlere özel komutları kullanmak için diğer değişken türlerini önce string türüne çevirmek gerekir. Aşağıda örnek verilmiştir.

pi=3.14159 #float tipinde bir değişken olduğu için örneğin kaç karakter olduğunu bulmak istersek hata #verecektir. O yüzden önce string türüne çeviririz.

a=str(pi) #str komutu ile a değişkeni pi değişkeninin string haline çevrildi.("3.1459")

- **String Birleştirme**

a="Kılıçarslan"

b="Lisesi"

c="Anadolu"

birleşik=a+c+b #değişken isimlerinin yanına + işareti koymamız metinleri birleştirir.

print (birleşik) #sonuç olarak "Kılıçarslan Anadolu Lisesi" çıkacaktır.

- **% İşareti İle Cümle Biçimlendirme**

isim="Zeynep"

print ("Merhaba %s"%(isim)) #sonuç olarak ekranda: Merhaba Zeynep yazacaktır.

- **Kullanıcıdan Bilgi Almak**

Python’da komut çalıştırıldığında kullanıcıdan cevap almak için input komutu kullanılır. Kullanımı aşağıdaki gibidir.

isim=input("İsminiz:") #isim değişkeninin değeri kod çalıştırıldığında kullanıcı cevap verince oluşur. input ile #gelen veri string türündedir.

## Karşılaştırma Operatörleri

Python'da Değişkenlerin ve Değerlerin karşılaştırılması için kullanılan işaretlere karşılaştırma operatörü denir. En sık kullanılanları aşağıda verilmiştir.

Eşittir	Eşit Değildir	Büyüktür	Küçüktür	Büyük Eşittir	Küçük Eşittir
==	!=	>	<	>=	<=

### • Tarih-Zaman Yazdırma

Güncel tarih ve saati yazdırmak için datetime fonksiyonu kullanılır.

```
from datetime import datetime
tarih=datetime.now()
print (tarih) #Güncel tarih ve saati yazdırır.
print (tarih.year) #Güncel yılı yazdırır.
print (tarih.month) #Güncel ayı yazdırır.
print (tarih.day) #Günü yazdırır.
print (tarih.hour) #Saati yazdırır.
print (tarih.minute) #Dakikayı yazdırır.
```

### • if Komutu

if komutu kodlamada şarta bağlı olarak gerçekleştirilmesi gereken komutlar olduğunda kullanılır. Örnekler aşağıda verilmiştir.

Kullanıcıya parola sorup doğru parola("deneme") yazıldığında ekrana "Hoş Geldiniz" yazısı çıkaran kod.

```
a=input("Parolayı Giriniz:")
if a=="deneme":
    print ("Hoş geldiniz") # if komutundaki şartın doğru(True) olmasına göre çalışan kodun 1 tab
                        #mesafesinde içeriden yazıldığını unutmayınız.
```

Kullanıcıya notunu sorup 50'den büyük ve 50'ye eşitse "geçti", 50'den küçükse "kaldı" yazan kod.

```
a=input("Notunuzu Giriniz:")
a=int(a)
if a>=50:
    print ("Geçti")
else:
    print ("Kaldı")
```

#input ile alınan veri string(metin) türünde olduğu için sayı ile karşılaştırılmaz.int komutu ile integer(tamsayı)'a dönüştürülür.

#else komutu if komutu ile belirlenen şartın gerçekleşmediği durumda yapılacakları belirtmek için kullanılır.

Kullanıcıya Notunu Sorup 0-50 arasına "Kaldınız",50-75 Arasına "Geçtiniz", 75-100 Arasına "Tebrikler" yazısı yazdırmak.

```
a=input ("Notunuzu Giriniz:")
```

```
a=int(a)
if a<=50:
    print ("Kaldı")
```

```
elif a<=75:
    print ("Geçti")
```

```
else:
    print ("Tebrikler")
```

# elif komutu ile birinci şartın gerçekleşmediği durumlarda ikinci bir şart yazılabilir. Bu da gerçekleşmezse else komutu içindeki komut çalışır.

Kullanıcıya Notunu Sorup Girdiği Not 0'dan Küçükse ya da 100'den büyükse uyarın kod.

```
a=input("Notunuzu Giriniz:")
```

```
a=int(a)
```

```
if (a>100 or a<0):
    print ("Notunuzu 0-100 arasında giriniz")
```

Bu örnekte olduğu gibi iki şarttan birinin gerçekleşmesinin yeterli olduğu durumlarda yapılacak işlemler için şartlar arasına or ifadesi yazılır.

Kullanıcıdan ad ve soyad bilgisi isteyip boş girmediğinde "Bilgileri Doğru Girdiniz" yazısını yazdırmak.

```
ad=input ("Adınızı Yazınız:")
```

```
soyad=input("Soyadınızı Yazınız:")
```

```
if (ad!="" and soyad!=""):
    print ("Bilgileri Doğru Girdiniz")
```

Örnekte olduğu gibi yazıyı yazmak için iki şartta aynı anda gerçekleşmesi gerekiyorsa şartlar arasına and ifadesi kullanılır.

### Örnekler:

1.Ekrana adınızı yazdırınız.

2.a=10,b=5 değişkenlerini kullanarak 4 işlem yapınız, sonuçlarını ekrana yazdırınız.

3.a=5 a değişkeninin 50.kuvvetini alıp ekrana yazdırınız.

4.a=100 b=3 a'nın b'ye bölümünden kalanı ekrana yazdırınız.

5.a="Monty Python" a değişkenini büyük ve küçük harflarla yazdırınız. Karakter sayısını yazdırınız. İlk 5 harfini yazdırınız.

6.Güncel tarih ve saati yazdırınız.

7.a="Ali",b="ata",c="bak." Değişkenlerini birleştirerek tek string haline getiriniz.

8.Kullanıcıdan bir sayı isteyip 10'dan küçükse ekrana "Rakam" yazdırınız.

9.Kullanıcıdan bir sayı isteyip sayı 0'dan büyükse "Pozitif", 0'dan küçükse "Negatif" yazdırınız.

10.Kullanıcıdan iki sayı isteyip büyük olanı küçükten çıkartıp ekrana yazdırınız.

Değerli öğrenciler, kâğıttaki bilgileri ve örnekleri ezberlemek yerine mantığını anlamaya çalışınız. Python dili çalışmak için telefonunuza ScodeApp uygulamasını yükleyebilirsiniz. Ayrıca kodlama denemeleri yapmak için Sololearn uygulamasını da yüklemenizi tavsiye ediyorum. Sololearn'de çalışırken Türkçe karakter kullanmayınız.