

1) print(1 + 6 + 7 + 10)

Yukarıdaki kod çalıştırıldığında sonuç olarak aşağıdaki şıklardan hangisi verilecektir?

a) 16410 b) Hata verir c) 24 d) 0

2) >>> a= int(6.39)
>>> b= int(6.99)
>>> c= a* b

Yukarıdaki işleme göre c=?

a) 30 b) 35 c) 36 d) 24

3) >>> ad= "hürkuş"
>>> soyad= 11
>>> ad+soyad

Yukarıdaki işlemin sonucu nedir?

a) hürkuş11
b) hürkuş+11
c) 66

d) 11 çift tırnak içerisine alınmadığı için metin değildir. Metin ile sayı toplanmaz(birleşmez) hata verir.

4) >>> y1,y2, perf=100,90,80
>>> ort=(y1+y2+perf)/3

Ort=? 90

5) >>> s=20

>>> s=s+1 Yandaki fonksiyonda programa s=s+1 satırını 10 kere döndüren program çalıştırıldığında s değişkeninin son değeri kaç olur?

a) 25 b) 10 c) 15 d) 30

6) x=10

x=x+5 Yandaki fonksiyonda programa x=x+5 satırını 5 kere döndüren program çalıştırıldığında x değişkeninin son değeri kaç olur?

a) 25 b) 45 c) 35 d) 30

7) z=2

z=z*2 Yandaki fonksiyonda programa z=z*2 satırını 4 kere döndüren program çalıştırıldığında z değişkeninin son değeri kaç olur?

a) 22 b) 32 c) 40 d) 20

Not: a=a+1 ifadesi a+=1 şeklinde de yazılabilir.

a=a*b ifadesi a*=b şeklinde de yazılabilir.

a=a/b ifadesi a/=b şeklinde yazılabilir.

8) >>> i = 8

>>> k = 6

i=? 48

>>> i*=k

k=? 54

>>> k+=i

Not: type(): verinin türünü öğrenmek için kullanılır.

int: tam sayı veri türünü ifade eder. Örn: int(4). Ondalık varsa virgülden sonrası atar. int(3.99) ifadesinin değeri 3 tür. Sayılar hiçbir zaman " " tırnak içerisine alınmaz. " " tırnak içerisine alındığı anda str türü anlamına gelir. Hata verir.

Float: Reel sayı

Str: Her türlü karakter dizisi, metin. Tırnak içerisinde ifade edilir. Örn: str("ömer") str("A4F16") parola=str("12345")

9) >>> type(4)

.....
>>> type("4")
.....

Yukarıdaki kod çalıştırıldığında noktalı yerlere sonuç olarak aşağıdaki şıklardan hangisi verilecektir?

a) <class "int">, <class "str"> b) <class "str">, <class "int">
c) <class "int">, <class "int"> d) <class "str">, <class "str">

Not: % işlemi bir bölümde kalanı verir. Örn: 13%4 = 1

// işlemi ise bir bölümde tam bölümü verir. Örn: 13//4 = 3

a**b işlemi ve pow(a,b) işlemi a sayısının b kuvvetini alır.

10) >>> 21%(13//2) işleminin sonucu nedir?

a) 1 b) 2 c) 3 d) 4

11) >>> 169**(120/240)

işlemin sonucu kaçtır?

a) 9.25 b) 13 c) 84.5 d) 26

12) >>> d=(5*(15-7)-2**3)/4-3

d=? 5

>>> e=pow(d-3,d)

e=? 32

13) >>> y=pow(80//6 - x , 2)

Verilen işlemin sonucunda y nin 25 değerini alabilmesi için x kaç olmalıdır?

a) 8 b) 9 c) 7 d) 5

14) >>> pow((89//4+x)%8, 2)

İşleminin sonucunun 25 olarak hesaplanabilmesi için x hangi değeri almalıdır?

a) 8 b) 7 c) 12 d) 9

15) >>> sayi1=int(input("1.sayıyı giriniz= "))

1.sayıyı giriniz=15

>>> sayi2=int(input("2.sayıyı giriniz= "))

2.sayıyı giriniz=25

>>> t= sayi1+sayi2

>>> print("Sonuç=",t)

Yukarıdaki programa göre t değeri kaçtır?

a) 15 b) 25 c) 40 d) 60

16) if ogrenciyas <18 OR ogrenciyas ==18 then

Yukarıdaki koşul aşağıdakilerden hangisinde doğru olarak tanımlanmıştır?

a) ogrenciyas değişkenindeki değer 18e eşit veya küçükse şart sağlanmış olur.

b) Ogrenciyas değeri 18den büyükse şart sağlanmış olur

c) Bu hatalı bir mantıktır program çalışmaz.

d) İstenilen şartlardan biri sağlanmaz ise koşul yerine

Not: = "eşittir" operatörü bir değişkene değer aktarmak için kullanılır.

Örn: a=0 , a değişkeninin değeri sıfırdır.

a=2 , a değişkeninin değeri sıfırken yeni bir değer aktararak 2 olmuştur.

Kodlamada eşitlik işlemi çift eşittir "==" kullanılır. Eşitliğin kontrolü if komutu gibi koşullu yapılarda kullanılır.

Örn: if a==2; eğer a değeri 2 ye eşit ise demek. Burada a değişkenine bir aktarma değil 2 ye olan eşitliği kontrol edilir eşit ise sonraki kodlar çalışır.

17) Aşağıdaki verilen operatör-görevi eşleştirmelerinden hangisi yanlıştır?

Operatör Görevi

a) ** Kuvvet, üs

b) >= Büyük veya eşit mi?

c) // Tam bölme

d) % Mod

e) = Eşit mi?

Not: **if**: eğer anlamına gelir. if den sonra yazılan koşul gerçekleştiği sürece altındaki komutlar çalışır. Koşul gerçekleşmez ise sonraki kodlar çalışmaya başlar. Ya da **else** ile koşul gerçekleşmediği durumda ne yapılacağını belirten komut yazılabilir.
input(): kullanıcıdan veri girmesini sağlamak ve girilen veri ile işlem yapmak için kullanılır.

```
18) sayi = int(input("Bir sayı giriniz: "))
if sayi > 10:
    print("Girdiğiniz sayı 10'dan büyüktür!")
if sayi < 10:
    print("Girdiğiniz sayı 10'dan küçüktür!")
if sayi == 10:
    print("Girdiğiniz sayı 10'dur!")
```

Yukarıdaki kodlara göre kullanıcı 7 değerini girdiğinde hangi sonuç ekrana çıkar?

- a) Girdiğiniz sayı 10'dan büyüktür!
- b) **Girdiğiniz sayı 10'dan küçüktür!**
- c) Girdiğiniz sayı 10'dur!
- d) Girdiğiniz sayı geçersizdir!

```
19)>>> kullanıcıAdi=input("Kullanıcı adınızı giriniz: ")
>>> sifre=input("Şifreni giriniz")
>>> if kullanıcıAdi=="admin" and sifre=="1234":
>>> Print("GİRİŞ BAŞARILI")
>>> else:
>>> Print("HATALI KULLANICI ADI YA DA ŞİFRE")
```

Verilen kodlar çalıştırıldığında; ;Kullanıcı Adı olarak "admin", şifre olarak "1234" girildiğinde ne olur?

- a) BÖYLE BİR KULLANICI ADI TANIMLI DEĞİL
- b) ŞİFRE DOĞRU AMA KULLANICI ADINIZI KONTROL EDİNİZ
- c) **GİRİŞ BAŞARILI**
- d) HATALI KULLANICI ADI YA DA ŞİFRE

20) Aşağıdaki önermelerden hangi "True" (Doğru) değeri döndürür?

- a) 45<30
- b) x=5, y=2, x+y==3
- c) **58>34**
- d) 35==25

```
21) sayi1=int(input("1.sayıyı giriniz: "))
sayi2= int(input("2.sayıyı giriniz: "))
sayi3= input("Hangi işlemi yapmak istersiniz + - * / :")
```

```
if işlem=="+":
    print("Sonuç: ", sayi1+sayi2)
if işlem=="-":
    print("Sonuç: ", sayi1-sayi2)
if işlem=="*":
    print("Sonuç: ", sayi1*sayi2)
if işlem=="/":
    print("Sonuç: ", sayi1/sayi2)
```

Yukarıda 4 işlem yapan kodlar çalıştıktan sonra kullanıcı klavyeden 25 ve 5 sayıları ile işlem için / karakteri girilirse işlem sonucu ne olur?

- a) 125
- b) 30
- c) **5**
- d) 20

```
22) sayi=int(input("BİR SAYI GİRİNİZ: "))
```

```
if sayi%2==0:
    print("Girilen sayı çifttir.")
else:
    print("Girilen sayı tektir.")
```

Yukarıdaki programa göre aşağıdaki seçeneklerden hangisi doğrudur?

- a) **Girilen sayının kalanına bakar sifıra eşit ise çifttir yazar**
- b) Girilen sayının kalanına bakar 1 e eşit ise tektir yazar
- c) Girilen sayının bölümüne bakar sifıra eşit ise çifttir yazar
- d) Girilen sayının bölümüne bakar 1 e eşit ise tektir yazar

```
23) galibiyet=int(input("Galibiyet Sayısı Giriniz: "))
beraberlik=int(input("Beraberlik Sayısı Giriniz: "))
atilan_gol=int(input("Atılan Gol Sayısı Giriniz: "))
yenilen_gol=int(input("Yenilen Gol Sayısı Giriniz: "))
puan=.....
averaj=.....
Print("PUAN = ", puan)
Print("AVERAJ= ", averaj)
```

Yukarıdaki programda galibiyet ve beraberlik sayılarına göre puan hesaplaması yapılmak istenmektedir.

Alınan her galibiyet 3, beraberlik 1 puan kazandırmaktadır. Puan hesaplanmasını sağlayan puan değişkeninin kodları ile boş bırakılmıştır.

A) Puan değişkeni formülü aşağıdakilerden hangisi olmalıdır?

- a) puan=(galibiyet+3)*(beraberlik*1)
- b) **puan=(galibiyet*3)+(beraberlik*1)**
- c) puan=(galibiyet*3)*(beraberlik*1)
- d) puan=(galibiyet+3)+(beraberlik*1)

B) Takımın averajını hesaplayan formül hangisidir?

Not: Averaj atılan gol ve yenilen golün farkıdır

- a) **averaj=atilan_gol-yenilen_gol**
- b) averaj=atilan-yenilen
- c) averaj=atilangol-yenilengol
- d) averaj=atilan_gol-yenilen

KOŞULLU DURUMLAR

1- İf-Else Yapısı

2- İf-Elif Yapısı

3- İf-Else-İf Yapısı

4- if-Elif-Else hepsinin bir arada kullanımı

1- if – else deyimi (Eğer – değilse / Koşullu İşlem yapma): Belirlenen şart doğruysa veya yanlışsa istenen işlemlerin yapılmasını sağlar. Birden fazla şart varsa **and** veya **or** bağlaçları kullanılır. Else (değilse) kullanımı zorunlu değildir. Kullanımı:

if (şart ya da şartlar):

Şart doğruysa yapılacak işlem (ler)

else:

Şart yanlışsa yapılacak işlem (ler)

Örnek 1: Aşağıda klavyeden girilen sayının 10 dan büyük, küçük veya eşit durumuna bakılarak kullanıcıya açıklama dönütü yapan kod

```
sayi = int(input("Bir sayı giriniz: "))
if sayi > 10: #Buradaki eğer diğer eğerlerden bağımsızdır
    print("Girdiğiniz sayı 10'dan büyüktür!")
if sayi < 10: #Buradaki eğer diğer eğerlerden bağımsızdır
    print("Girdiğiniz sayı 10'dan küçüktür!")
if sayi == 10: #Buradaki eğer diğer eğerlerden bağımsızdır
    print("Girdiğiniz sayı 10'dur!")
```

Açıklama: **input** fonksiyonu ile kullanıcıdan sayı girilmesi sağlanır. Girilen değer **int()** içerisine alınarak virgülden sonraki kısımlarını atar tam sayıya dönüştürür. Girilen bu sayı **sayi** değişkenine atanır. İf koşul yapısı ile **sayi** değişkeni 10 dan büyük, 10 dan küçük yada 10 eşit durumu kontrol edilerek sağlanan koşula göre **print(" ")** komutu ile kullanıcıya(ekrana) açıklama yazan kod.

Örnek 2: Girilen nota göre ekrana geçtiğini/kaldığını yazan kod

```
ortalama=int(input("Not ortalamanızı giriniz:"))
if (ortalama>=50):
    print("Geçtiniz")
else:
    print("Kaldınız")
```

Açıklama: **input** fonksiyonu ile kullanıcıdan not girilmesi sağlanır. Girilen değer **int()** içerisine alınarak virgülden sonraki kısımlarını atar tam sayıya dönüştürür. Girilen bu sayı **ortalama** değişkenine atanır. İf koşul yapısı ile **ortalama** değişkeni 50 ye eşit ve büyükse **print()** komutu **geçtiniz**, **else** yani 50 ye ve büyük değilse **kaldınız** açıklama yazan kod.

2- Elif (Else İf – Değilse Eğer): Elif deyimi else if 'in kısaltmasıdır. İç içe eğerler oluşturularak istenen sayıda şart kontrol edilebilir. Elif ifadeleri en baştaki if ifadesine bağlıdır. İf ve if'e bağlı elif deyimleri aynı hizadadırlar!

Örnek 1: Kullanıcının yaşına göre açıklama yazan kod:

```
yas = int(input("Yaşınız: "))
if yas == 18: #Eğer yas içindeki sayı 18'e eşitse ?
    print("Yeni reşit oldunuz")
elif yas < 0: # Değilse, eğer yas içindeki sayı 0'dan küçükse?
    print("Yaş sıfırdan küçük olamaz")
elif yas < 18: # Değilse, eğer yas içindeki sayı 18'den küçükse?
    print("Genç bir kardeşimizsin!")
```

elif yas > 18: # Değilse, eğer yas içindeki sayı 18'den büyükse?

```
print("Yaş yavaş yavaş kemale eriyor!")
```

Açıklama: **input** fonksiyonu ile kullanıcıdan yaşı girilmesi sağlanır. Girilen değer **int()** içerisine alınarak virgülden sonraki kısımlarını atar tam sayıya dönüştürür. Girilen bu sayı **yas** değişkenine atanır. **İf** ile yas değeri 18 e ise “yeni reşit oldunuz” açıklamasını ekrana yazar. Girilen değer 18 e eşit değilse bir sonraki koşul **elif** yapısı ile 0 dan küçük olma durumu kontrol edilerek eğer 0 küçük ise “Yaş sıfırdan küçük olamaz” yazısı ekrana yazar. Diğer **elif** yapılarında da 18 den küçük-büyük olma durumuna bakılarak kullanıcıya açıklama yazan kod.

3- İç içe if-else yapısı: Eğer çok sayıda koşula göre farklı işlemler yaptırılmak isteniyorsa iç içe if-else yapısı kullanılabilir. Girintilere dikkat etmek gerekir(Girintiler, if yapılarının birbirine bağlı olduklarını gösterir). Ancak kullanım kolaylığı açısından **elif** yapısını kullanmak daha basittir.

Örnek 1: Nota göre dereceyi yazan program:

```
dNotu=int(input("Ders notunu gir:")) # Tamsayı bilgi giriliyor
```

```
if dNotu>=85 and dNotu<=100:
```

```
    print("Pekiyi")
```

```
else:
```

```
    if dNotu>=70 and dNotu<85:
```

```
        print("İyi")
```

```
    else:
```

```
        if dNotu>=60 and dNotu<70:
```

```
            print("Orta")
```

```
        else:
```

```
            if dNotu>=50 and dNotu<60:
```

```
                print("Geçer")
```

```
            else:
```

```
                if dNotu>=0 and dNotu<50:
```

```
                    print("Geçmez")
```

```
                else:
```

```
                    print("Geçersiz not girdiniz!!")
```

Açıklama: **input** fonksiyonu ile kullanıcıdan ders notu girilmesi sağlanır. Girilen değer **int()** içerisine alınarak virgülden sonraki kısımlarını atar tam sayıya dönüştürür. Girilen bu sayı **dNotu** değişkenine atanır. **İf** ile girilen notun 85 e eşit ve büyük ve **and operatörü** ile 100 e eşit ve küçük olma koşul durumu bakılarak eğer koşullar gerçekleşirse kullanıcıya **print** ile Pekiyi açıklaması yazdırılır. Girilen not ilk koşullara uygun **değilse** (1.**else**) 2. **if** ile **dNotu** değişkenindeki değer 70 e eşit ve büyük ve 85 den küçük koşul durumu sağlanıyorsa “iyi” açıklaması ekrana yazar. Eğer sayı bunlara da uymuyorsa(**2.else**) **3.if** ile 60 dan büyük ve eşit ve 70 den küçük olma durumu kontrol edilir. Eğer sayı bu aralıkta ise “orta” açıklaması yazar. Yine değilse **3.else** ve **4.if** ile 50 den büyük ve eşit ve 60 dan küçük olma durumuna göre “geçer” yazar ekrana. Bu aralıkta da sayı **4.else** ve **5.if** ile notun 0 a eşit ve büyük ve 50 küçük olma koşulu ile ekrana “geçmez” yazar. Hiçbir koşula uymuyorsa **5.else** ile “Geçersiz bir not girdiniz!!” açıklaması yazar ekrana.

4- if – elif ve else deyimlerinin birlikte kullanımı: Elif deyimi tek başına kullanılmaz. Eğer (if) ile karar işlemleri başlatılır. Birden fazla karar ifadesi varsa elif deyimleri ile karar aldırma işlemi yapılır. Eğer bu karar deyimlerinden hiçbiri doğru değilse “else” deyimi kullanılıp bu durumda çalıştırılacak komutlar belirlenebilir.

Örnek 1: Not ortalamasına göre alınan belgeyi ekrana yazdıran program:

```
notunuz=int(input("Not ortalamanızı giriniz:"))
```

```
if notunuz>=85 and notunuz<=100:
```

```
    print("Takdir aldınız")
```

```
elif notunuz>=70 and notunuz<85:  
    print("Teşekkür aldınız")  
elif notunuz>=50 and notunuz<70:  
    print("Belge alamadan geçtin")  
elif notunuz>=0 and notunuz<50:  
    print("Kaldınız")  
else: #Belirtilen şartlardan hiçbiri doğru değilse  
    print("Geçerli bir not girmediniz!")
```

Açıklama: **input** fonksiyonu ile kullanıcıdan ders notu girilmesi sağlanır. Girilen değer **int()** içerisine alınarak virgülden sonraki kısımlarını atar tam sayıya dönüştürür. Girilen bu sayı **notunuz** değişkenine atanır. **If** ile girilen notun 85 e eşit ve büyük ve **and** operatörü ile 100 e eşit ve küçük olma koşul durumu bakılarak eğer koşullar gerçekleşirse kullanıcıya **print** ile "Takdir aldınız" açıklaması yazdırılır. Girilen not ilk koşullara uygun **değilse 1.elif** ile **notunuz** değişkenindeki değer 70 e eşit ve büyük ve 85 den küçük koşul durumu sağlanıyorsa "Teşekkür aldınız" açıklaması ekrana yazar. Eğer sayı bunlara da uymuyorsa **2.elif** ile 50 dan büyük ve eşit ve 70 den küçük olma durumu kontrol edilir. Eğer sayı bu aralıkta ise "Belge alamadan geçtin" açıklaması yazar. Yine değilse **3.elif** ile 0 a eşit ve büyük ve 50 küçük olma koşulu ile ekrana "Kaldınız" yazar. Hiçbir koşula uymuyorsa **else** ile "Geçerli bir not girmediniz!" açıklaması yazar ekrana.

DÖNGÜ YAPILARI

1. For Döngüsü
2. While Döngüsü
3. Döngüden çıkış(break)
4. Continue(devam)

1. **For Döngüsü:** Belirtilen işlem veya işlemleri **istenilen (Belirli) sayıda** tekrarlamak için kullanılır.

Kullanımı: **for** döngü değişkeni **in** liste ya da sayı aralığı:

Örnek 1:	Ekran çıktısı
kelime="Deneme"	D
for karakter in kelime:	e
print(karakter)	n
	e
	m
	e

Açıklama: kelime değişkenine "Deneme" yazısı(karakter dizisi) atanmıştır ve 6 harften oluşur. **For** döngüsü **karakter** döngü **değişkenine kelime** dizisinin her bir harfini sıra ile atar. **Print** ile döngü her döndüğünde **karakter** döngü değişkenine atanan harf ekrana yazdırılır. 6 harf-karakterden oluştuğu için 6 kere döngü döner.

For döngüleri başlangıç ve bitiş değerleri arasında artım miktarına göre istenilen sayıda tekrar yapar. Bunu **range** komutu kullanılır.

range (başlangıç değeri, son değeri, arttırma/azaltma değeri) :

Başlangıç değeri: Döngü değişkeninin alacağı ilk değerdir. Eğer boş bırakılırsa 0 olarak belirlenir.

Son değeri: Döngü değişkeninin bitiş değeridir. Boş bırakılmamalıdır.

Arttırma/azaltma değeri: Döngü değişkeninin artırma veya azaltma miktarını belirler. Eğer boş bırakılırsa, 1 olarak belirlenir. Başlangıç, bitiş, artırma ve azaltma değerlerinin hepsi tam sayı olmalıdır. Ondaklıkları değerler veya diğer veri türleri kullanılmaz. Başlangıç değeri dahil, bitiş değeri dahil değildir!!

Örnek 1: 1 den 5'e kadar sayıları ekrana yazdıran program:

for sayi **in** range(1,6,1): # 5 i yazdırması için 6 verildi.

```
print(sayi)
```

Ekran Çıktısı:

```
1
2
3
4
5
```

Açıklama: for döngü yapısı ile sayi döngü değişkenine 1 den başlayıp 1 artırarak 5 e kadar olan sayılar atanır. Döngü her döndüğünde print komutu ile sayi değişkenindeki değer ekrana yazdırır. 5 kere döngü döner. 1,2,3,4,5 yazar ekrana.

Örnek 2: 1'den 100'e kadar tek sayıları yazdıran program:

for sayi **in** range(1,100,2):

```
print(sayi) # 1 den 99 a kadar tek sayıları yazdırır
```

Ekran Çıktısı:

```
1
3
5
7
...
99
```

Açıklama: 1 den başlayıp 2 artırarak 99 a kadar olan sayıları tek tek ekrana yazar. Sonuçta ekranda 1,3,5,7,...,99 tek sayıları yazılmış olur.

2. WHILE Döngüsü: İstenen komutların **belirli** ya da **belirsiz** sayıda tekrar etmesi için kullanılır. Belirtilen şart doğru olduğu sürece istenen komut/komutları tekrar eder.

Kullanımı: while şart/şartlar:

```
komut1
komut .....
```

Örnek 1:

Negatif sayı girilinceye kadar girilen sayıları toplayan while döngüsü (Bu döngünün kaç kez döneceği **belirsizdir**. Çünkü kullanıcıya bağlıdır. Kullanıcı negatif bir sayı girdiğinde döngüden çıkarılır)

sayi,toplam = 0,0 # sayi ve toplam değişkenlerine ilk değer olarak 0 ver.

```
print("Bir sayı giriniz, negatif sayı döngüyü sonlandırır:")
```

```
while sayi >= 0: #sayi değeri 0'a eşit ya da büyük olduğu sürece tekrarla
```

```
    sayi= int(input("Sayı giriniz:"))
```

```
    if sayi>0:
```

```
        toplam+=sayi #toplam=toplam+sayi
```

```
print("Toplam=", toplam)
```

Açıklama: **sayi** ve **toplam** isimli 2 değişkene başlangıç değeri olarak **sıfır** verilir. Çünkü program içerisinde bu değişkenler değişiklik gösterecek program her yeniden başlatıldığında sıfır değeri bu şekilde atanarak işlemin sonucunun doğru çıkması sağlanmış olacaktır. **Print** ile kullanıcıya açıklama gösterilir. **While** döngüsünde şart olarak sayi değişkenine girilen değerin **sıfıra eşit ve büyük** olması yazılmıştır. Sıfırdan küçük bir değer while döngüsüne girmez ve döngü sonlanır. **Sayi** değişkeninin ilk değeri sıfır olduğu için döngü çalışır **input** ile kullanıcıdan sayı girilmesi istenir. Girilen sayının tam sayı olması için **int** kullanılır. **Toplam** sayacının doğru çıkması girilen negatif değerlerin toplama aktarılmaması için **if** ile **sayi** nin **sıfırdan büyük** olması kontrol edilir. Girilen pozitif sayılar **toplam+=toplam+sayi** denklemi ile toplam değişkeninde biriktirilir(üst üste toplanır). Negatif değer girilip döngü sonlandığında kullanıcıya **print** komutu ile **"Toplam="** açıklaması en son **toplam** değer ekrana yazdırılır.

3. **Döngüden Çıkış (break)** : Döngü içindeyken belirli bir şart sağlandığında break komutu ile çıkış yapılması (döngünün sonlanması) sağlanabilir. For veya While döngüsünde kullanılabilir.

Negatif sayı girilinceye kadar sayıları toplayan program:

```
giris,toplam = 0,0
while True: # Sonsuza kadar dön! Çünkü şart True yani "Doğru"
    giris = int(input("Sayı giriniz:"))
    if giris < 0: # Eğer giriş değişkeni değeri 0 dan küçükse
        break # Döngüden çık
    toplam += giris
print("Toplam =", toplam) # Döngüden çıkınca yazma işlemi yap
```

4. **Continue:** Break döngünün çalışmasını sonlandırıp, döngüden sonraki komutlara geçilmesini sağlarken, Continue deyimi döngüden çıkılmadan **döngü başı** yapılarak, bir sonraki yeni değer için işlem yapılmasını sağlar.

Girilen pozitif sayıların toplamını hesaplayan program. Negatif sayılar Continue deyimi ile işleme alınmamaktadır. Programdan çıkmak için 999 girmek gerekir.

```
toplam,girilen = 0,0 #toplam ve girilen değişkenlerine 0 değerini ata.
while girilen!=999: # girilen değeri 999 olmadığı sürece tekrar et!
    girilen = int(input("Pozitif sayı giriniz (Çıkış için 999):"))
    if girilen < 0: # girilen değer 0 dan küçükse
        print("Değer işleme alınmadı") # yaz
    continue # Döngü başı yap (Aşağıdaki kodu çalıştırmadan)
if girilen != 999:
    toplam += girilen
    print("Toplam =", toplam) # Döngüden çıkınca toplam değerini yazdır
```