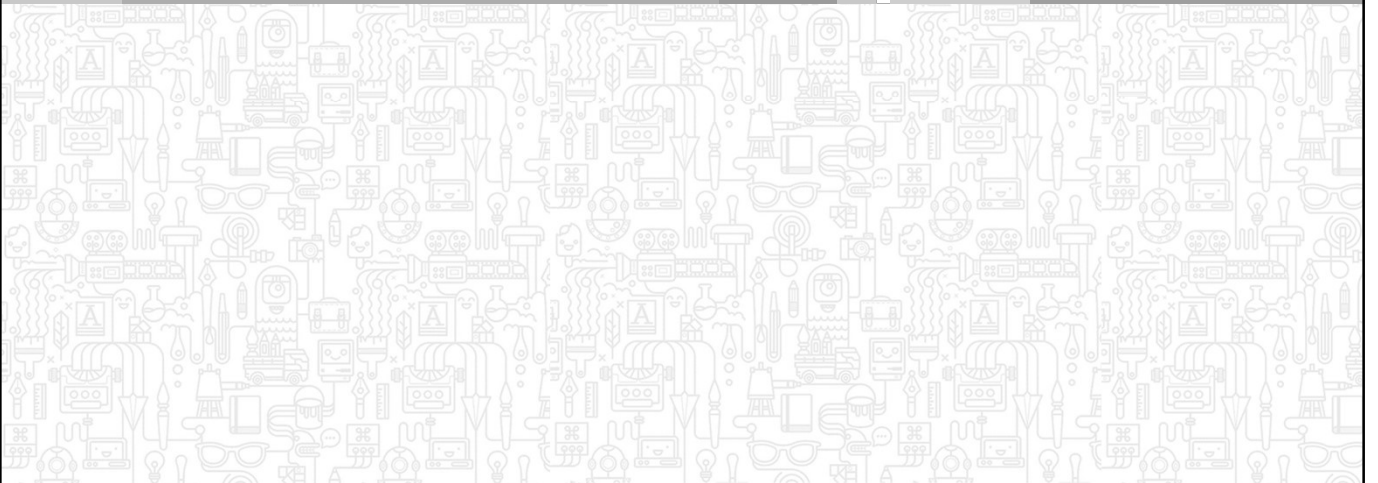


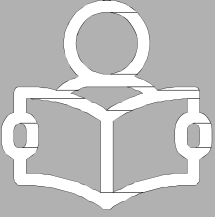
PYTHON



**Koşullu ve Mantıksal İfadeler**



## Kapsam



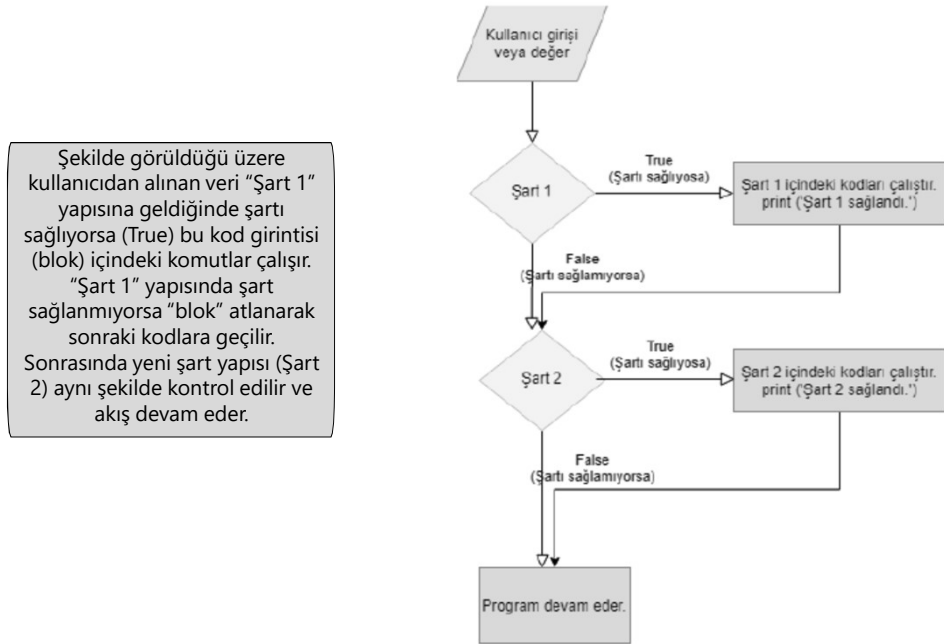
- ✓ Koşullu İfadeler
- Mantıksal İfadeler ve Bağlaçlar
- Python Blok Yapısı
- if yapısı
- İç İçe Koşul İfadeleri
- if-else Yapısı
- if-elif-else Yapısı

## Koşullu İfadeler

Program yazarken kodlar sıralı olarak alt alta satırlar şeklinde yazılır. Programın akışında bazı dallanmalar (farklı durumlar için farklı kodların çalışmasını istediğimiz durumlar) olabilir. Programlarda belirli koşul veya durumlar için çalışması istenen kodlar koşullu ifade blokları kullanılarak oluşturulur.

- ✓ Örneğin, sadece üyelerin giriş yapabileceği bir program hazırlarken kullanıcı adı ve parolası doğru olan kişilerin sisteme erişimine izin veren kodlar yazılır. Bunun için programa koşul ifadeleri eklenmesi gerekir. Koşul sağlanıyorsa menülere erişim izni verecek kodlar çalışmaya başlar. Koşul sağlanmıyorsa uyarı mesajı verilerek sisteme giriş izni verilmez.
- ✓ Mantıksal operatörler sonuç olarak "boolean" veri tipinde değer verir. Eğer koşul sağlanırsa "True" değeri döndürürken koşul sağlanmazsa "False" değeri döndürür. "Boolean" veri tipi bu iki değerden başka bir değer alamaz. Bu durum koşullu ifadeler üretme olanağı sağlar. Koşullu ifadelerin sonucu "boolean" değer kontrol edilerek program akışı yönlendirilebilir.

## Koşullu İfadeler



## Mantıksal İfadeler ve Bağlaçlar

İlişkisel operatörler ve mantıksal bağlaçlar kullanılarak (or, and, not gibi) farklı koşul durumları oluşturulur.

### Örnek 1

Kod

```

kullaniciAdi=input('Kullanıcı Adı: ')
kullaniciParola=input('Parola: ')
print(kullaniciAdi=='Admin' and kullaniciParola=='123456')
  
```

Örnek kodda kullanıcı adı ve şifresi doğru girilirse ekrana "True" ikisinden biri bile yanlış olursa "False" değeri döndürür.

## Python Blok Yapısı

Python'da kodlar belirli alt kümeler (bloklar) halinde ifade edilir.

Bu yapı Python'da girinti (4 space veya 1 tab) ile oluşturulur.

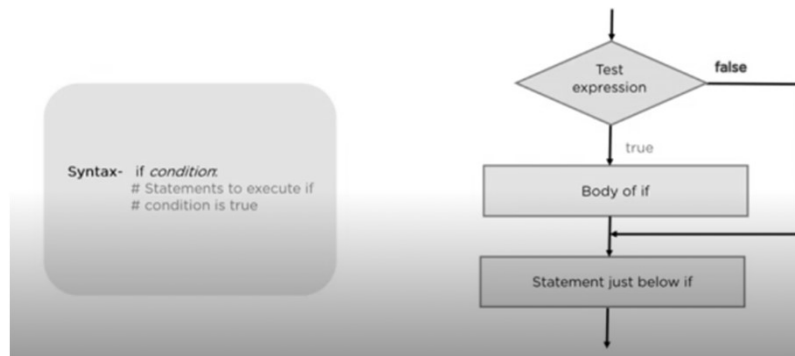
Python'da dikey olarak aynı hizadaki kodlar aynı blok yapısında yer alır.

Döngüler, fonksiyonlar ve koşul ifadeleri kod blokları kullanılarak oluşturulur.

```
def isyukuHesapla():
    for m in rangeModel:
        cycle[m]=0
    toplam=0
    for s in rangeIsci:
        for m in rangeModel:
            stn=m*isci-(isci-1)
            for j in rangeIsci:
                for i in rangeGorev:
                    if ata[i]==s and ataisci[j]==s:
                        toplam=toplam+pt.loc[i,stn]
            stn=stn+1
        isyuku.loc[s,m]=toplam
    if cycle[m]<toplam: cycle[m]=toplam
    toplam=0
    isyukuHesapla()
```

## if yapısı

Bu yapıda, belirli komutların çalışması, bir koşula bağlıdır. Koşul sağlanmazsa bu komutlar çalışmaz.



Koşul karşılaştırma sonucu "boolean" (True veya False) bir değer verir. True değer verirse "if" bloğu içinde (girintide olan) kodlar çalışır. Koşul sağlanmazsa yani "False" değeri verirse bloğun içine girilmez bloktaki kodlar atlanır.

## if yapısı

Kullanıcının yaş değerini alarak 18'e eşit veya büyük olması hâlinde ona mesaj veren kod:

### Örnek 2

Kod

```
yasi=int(input('Lütfen yaşıınızı giriniz: '))
if (yasi>=18):
    print('Oy kullanabilirsiniz.')
print ('Program bitti.')
```

## if yapısı

Koşul ifadelerinde birden fazla koşul birlikte kullanılabilir. Bunun için mantıksal operatörler kullanılır. Örnekte "and" operatörü ile iki koşulun birlikte sağlanma şartı koşulmuştur.

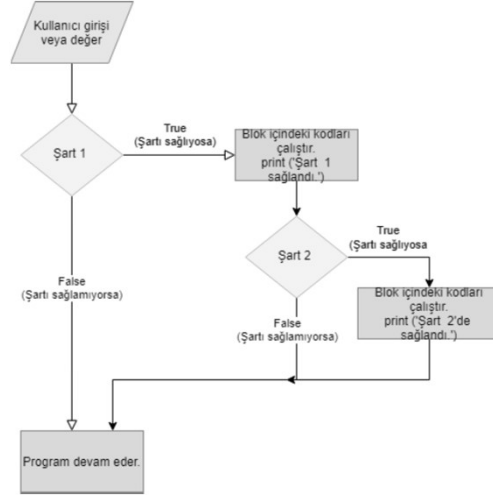
### Örnek 3

Kod

```
kullaniciAdi=input('Kullanıcı Adı: ')
kullaniciParola=input('Parola: ')
if (kullaniciAdi=='Admin' and kullaniciParola=='123456'):
    print('Giriş başarılı.')
    print ('Menülere erişebilirsiniz.')
print('Kullanıcı adı veya parola yanlış.')
```

## İç İçe Koşul İfadeleri

Bu yapıda, birden fazla iç içe koşul ifadesi bulunmaktadır.



Bu kullanım bir önceki örnekteki "and" operatörü kullanımına benzetmekle birlikte koşullardan sadece birincisinin sağlandığı durumlar için kodlar oluşturmaya olanak verir. İç içe koşul ifadelerinin sayısı (iç içe 3 koşul, 5 koşul gibi) ihtiyaca göre artırılabilir.

## İç İçe Koşul İfadeleri

Koşul ifadelerinde birden fazla koşul birlikte kullanılabilir. Bunun için mantıksal operatörler kullanılır. Örnekte "and" operatörü ile iki koşulun birlikte sağlanma şartı koşulmuştur.

### Örnek 4

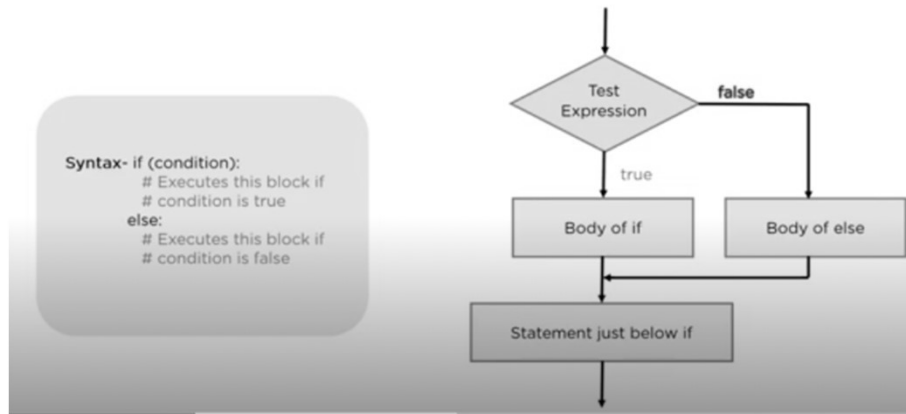
Kod

```

kullaniciAdi=input('Kullanıcı Adı: ')
if (kullaniciAdi=='Admin'):
    kullaniciParola=input('Kullanıcı Adı doğru. Lütfen Parola'yı giriniz: ')
    if (kullaniciParola=='123456'):
        print('Giriş başarılı.')
        print('Menülere erişebilirsiniz.')
  
```

## if-else Yapısı

"if" yapısında şart sağlanırsa blok içindeki kodlar çalışmaktadır. Ancak şartın sağlanmadığı durumlarda herhangi bir işlem yapılmaz. "else" ifadesi değilse anlamındadır. Yani şartın sağlanmadığı durumda çalışacak kodlar "else" bloğuna yazılır.



"else" bloğu da "if" bloğu gibi ayrı bir blok olarak yazılır. Bir "if" bloğundan sonra gelen "else" bloğu aynı girinti seviyesinde olmalıdır. "else" bloğu, "if" ile birlikte kullanılır.

## if-else Yapısı

Aşağıdaki örnekte kullanıcının girdiği sayının çift - tek olduğunu bulan bir kod yazılmıştır.

Çift bir sayı girildiğinde "if" bloğunun içindeki kodlar çalışır. Tek bir sayı girildiğinde "else" bloğunun içindeki kodlar çalışır.

### Örnek 5

Kod

```

sayi1=int(input('Lütfen bir sayı giriniz: '))
if sayi1%2==0:
    print('Girdiğiniz sayı çifttir.')
else:
    print('Girdiğiniz sayı tektir.')
  
```

## if-else Yapısı

Koşullu ifade, operatörler ve bağlaçlarla daha etkili yapılabilir. Az önceki kullanıcı adı ve parolası örneğinde kullanıcının girdiği kullanıcı adı ve parola bu örnekte "and" ile birlikte kontrol edilmiştir.

### Örnek 6

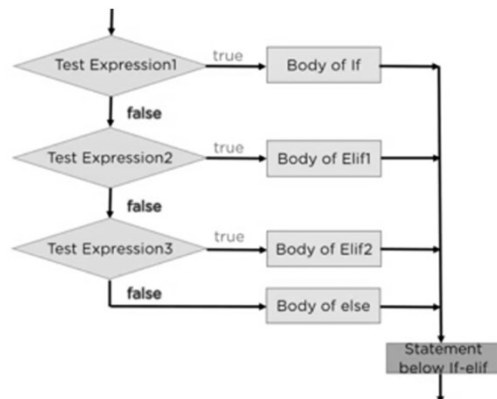
Kod

```
kullaniciAdi=input('Kullanıcı Adı: ')
kullaniciParola=input('Parola: ')
if (kullaniciAdi=='Admin' and kullaniciParola=='123456'):
    print('Giriş başarılı.')
    print ('Menülere erişebilirsiniz.')
else:
    print ('Yanlış kullanıcı adı veya şifre')
```

## if-elif-else Yapısı

Bu yapıda koşullar art arda verilir. if ile verilen koşulun devamında 'değilse şu ise' anlamına gelen "elif" ifadesi yer alır. Yapının en sonunda ise 'hiçbiri değilse' anlamında else ifadesi yer almaktadır. Her ifade kendi bloğundaki kodları çalıştırır. "if, elif ve else" bloklarının girinti düzeyleri aynı olmalıdır.

**Syntax-** if (condition):  
statement  
elif (condition):  
statement  
.  
.  
else:  
statement



Bir değerin belirli aralıktaki şartları sağlayıp sağlamadığı kontrol edilirken "if-elif-else" yapısını kullanmak daha uygundur. Bu yapıda koşullardan biri sağlanıyorsa diğer koşullar kontrol edilmez. Alınan değer "if-elif-else" yapısındaki yalnız bir koşulu sağlayabilir.



## if-else Yapısı

Bir kullanıcının sınav puanını alarak durumunun değerlendirilmesi:

### Örnek 7

Kod

```
sinavPuanı=int(input('Puanınız giriniz (0-100): '))
if sinavPuanı>=85:
    print('Pek iyi')
elif sinavPuanı>=70:
    print('İyi')
elif sinavPuanı>=55:
    print('Orta')
elif sinavPuanı>=45:
    print('Geçer')
else:
    print('Kaldı')
```