

C# KONTROL YAPILARI

Adem AKKUŞ
Bilgisayar Mühendisi
Bilişim Teknolojileri Öğretmeni
Eğitmen

Ankara, 2023



KONTROL YAPILARI

Şu ana kadar yapmış olduğumuz işlemlerin gerçekleştirilmesinde herhangi bir şart aranmıyordu.

Herhangi bir şart doğru olduğunda gerçekleştirmek isteyeceğimiz durumlar olabilir.



KONTROL YAPILARI

Örneğin kullanıcıdan 0 ile 100 arasında bir sayı girmesini istiyorsunuz.

Burada kullanıcının sizin istediğiniz gibi 0-100 arası bir sayı gireceğiniz garanti edemezsiniz.



KONTROL YAPILARI

Yapmanız gereken şey istediğiniz aralıkta bir sayı girmediğinde yeni bir sayı girmesini sağlamaktır.

Burada şartınız kullanıcının girdiği sayının 0 dan büyük ve 100 den küçük olmasıdır.



KONTROL YAPILARI

Kontrol ifadeleri programlama dilinin temelini oluşturur.

C# en sık kullanılan kontrol yapısı **if-else** yapısıdır.

Diğer bir kontrol ifadesi de switch-case yapısıdır.



if-else YAPISI

```
if (kontrol ifadesi)
{
    işlemler;
}
else
{
    işlemler;
}
```



if-else YAPISI

```
if (koşul)
{
    işlemler;
}
else
{
    işlemler;
}
```

if içerisine yazılan koşul mutlaka parantez içerisinde yazılmalı.

Yazılan koşul mutlaka mantıksal bir koşul olmalıdır.

Yani ya yanlış (false) ya da doğru (true) sonucu üretmelidir.



if-else YAPISI

```
int a=5;  
if (a=5)  
{  
    işlemler;  
}
```

a=5
İfadesinde = operatörü atama
operatörüdür.
Mantıksal sonuç üretmez.

Doğrusu
if(a==5)
{

}



if-else YAPISI

```
if (kontrol ifadesi)
{
    işlemler1;
}
else
{
    işlemler2;
}
```

Kontrol ifadesinde belirtilen şartın sağlanıp sağlanmadığına bakılır.

Eğer şart sağlanıyorsa ilk blok işletilir ve işlemler1 deki komut veya komutlar çalıştırılır.

Şart sağlanmıyorsa else bloğu çalıştırılır.

Hiçbir zaman iki blokta çalışmaz. Çünkü if ile belirtilen şart ya sağlanıyordur yada sağlanmıyordur.



if-else YAPISI

```
if (1==1)
```

```
    Console.WriteLine("Şart Sağlanıyor");
```

```
else
```

```
    Console.WriteLine("Şart Sağlanmıyor");
```

Eğer if ve else blokları içerisinde bir komut işletilecekse o zaman süslü parantez kullanmaya gerek yoktur.



if-else Örnek 1

```
//kontrol sonucu olarak boolean bir değer verir:True ya da False
//kullanıcı tarafından girilen sayının Negatif ya da Pozitif olduğunu program
int sayi;
Console.Write("Bir sayı giriniz:");
sayi=Convert.ToInt32(Console.ReadLine());
if (sayi < 0)
{
    Console.WriteLine("Sayı Negatif.");
}
else
{
    Console.WriteLine("Sayı Pozitif.");
}
```



if-else Örnek 2

Klavyeden girilen bir sayının tek mi, çift mi olduğunu bulan c# kodlarını yazınız?

```
int sayi;  
Console.Write("Sayıyı Gir: ");  
sayi = Convert.ToInt16(Console.ReadLine());  
  
if (sayi%2==0)  
{  
    Console.Write("Bu Sayı ÇİFT Sayıdır");  
}  
else  
{  
    Console.Write("Bu Sayı TEK Sayıdır");  
}  
Console.ReadKey();
```



if-else if-else YAPISI

Eğer if şartı sağlanmadığında gerçekleşecek olan else durumunu da kendi içerisinde özel şartlara bağlamak istiyorsak o zaman if-else if-else yapısı kullanılır.



if-else if-else YAPISI

if (koşul)

```
{  
Deyimler1;  
}
```

else if(koşul2)

```
{  
Deyimler2;  
}
```

else

```
{  
Deyimler3;  
}
```

if şartındaki koşul sağlanırsa
Deyimler1 ifadeleri çalıştırılır.

Eğer if koşulu sağlanmıyorsa o
zaman else if de yer alan koşul
kontrol edilir.

O da sağlanmıyorsa o zaman else
deyimindeki ifadeler gerçekleştirilir.



if-else if-else YAPISI

if (a==b)

```
{  
Console.WriteLine("a=b");  
}
```

else if(a<b)

```
{  
Console.WriteLine("a<b");  
}
```

else

```
{  
Console.WriteLine("a>b");  
}
```

a değişkeninin değeri 10, b değişkeninin değeri ise 20 olsun. if koşulunda $a==b$ ifadesi sağlanmıyor.

Bu ifadenin değili $a<b$ veya $a>b$ olabilir.

Bunlardan $a<b$ şartı sağlanıyorsa küçük sayı a, $a>b$ sağlanıyorsa o zaman küçük sayı b yazdıralım.



if-else if-else YAPISI

```
if (a==b)
```

```
{  
    Console.WriteLine("a=b");  
}
```

```
if (a<b)
```

```
{  
    Console.WriteLine("a<b");  
}
```

```
if (a>b)
```

```
{  
    Console.WriteLine("a>b");  
}
```

Eğer else if yapısı kullanılmazsa ayrı ayrı if kullanılmalıdır.

Bu da önceki else if uygulamasından farklı çalışmaz.

Ancak bu programda else-if kullanırsak okunurluk artar.



if-else if-else Örnek 1

Klavyeden girilen sayının , Negatif Pozitif veya Sıfıra eşit olup olmadığını bulan programı yazınız.



if- else if- else Örnek 1

```
//kontrol sonuç olarak boolean bir değer verir:True ya da False
//kullanıcı tarafından girilen sayının Negatif ya da Pozitif
//ya da Sıfır olduğunu program
int sayi;
Console.WriteLine("Bir sayı giriniz:");
sayi=Convert.ToInt32(Console.ReadLine());
if (sayi < 0)
{
    Console.WriteLine("Sayı Negatif.");
}
else if (sayi > 0)
{
    Console.WriteLine("Sayı Pozitif.");
}
else
{
    Console.WriteLine("Sayı SIFIR.");
}
```



if- else if- else Örnek 2

Klavyeden girilen bir sayının tek mi, çift mi olduğunu bulan c# kodlarını yazınız?



if-else if-else Örnek 3

Girilen kenar uzunluklarına göre bir üçgenin türü hakkında bilgi veren C# programını yazınız(Mantıksal operatörler ile yapınız)



if-else if-else Örnek 3

Girilen kenar uzunluklarına göre bir üçgenin türü hakkında bilgi veren C# program

```
Console.Write("A kenar Uzunluğunu Giriniz:");  
int A = Convert.ToInt32(Console.ReadLine());  
Console.Write("B kenar Uzunluğunu Giriniz:");  
int B = Convert.ToInt32(Console.ReadLine());  
Console.Write("C kenar Uzunluğunu Giriniz:");  
int C = Convert.ToInt32(Console.ReadLine());  
  
if (A == B && B == C)  
{  
    Console.WriteLine("Eşkenar Üçgen");  
}  
else if (A == B || B == C || A == C)  
{  
    Console.WriteLine("İKİZKENAR ÜÇGEN");  
}  
else  
{  
    Console.WriteLine("Çeşitkenar Üçgen");  
}
```



if-else if-else Örnek 3

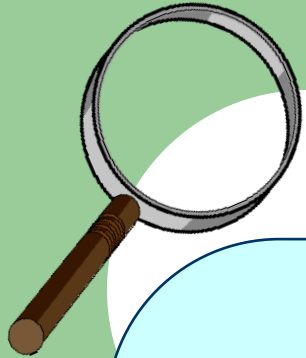
```
//üçgen çeşitleri
byte a, b, c;
Console.Write("a=");
a = Convert.ToByte(Console.ReadLine());
Console.Write("b=");
b = Convert.ToByte(Console.ReadLine());
Console.Write("c=");
c = Convert.ToByte(Console.ReadLine());
if (a == b && b == c)
{
    Console.WriteLine("Eşkenar üçgen");
}
else if (a == c || b == a || b == c)
{
    Console.WriteLine("ikiz kenar üçgen");
}
else
{
    Console.WriteLine("Çeşitkenar üçgen");
}
```



if-else if-else Örnek 4



```
//öğrenci notu girilecek:vize,final,proje
//vize %30,final:%50, proje %20 ortalaması alınacak .
//geçer not 60
double vize, final, proje, ortalama;
Console.Write("Vize notunu giriniz:");
vize=Convert.ToDouble(Console.ReadLine());
Console.Write("Final notunu giriniz:");
final=Convert.ToDouble(Console.ReadLine());
Console.Write("Proje notunu giriniz:");
proje=Convert.ToDouble(Console.ReadLine());
ortalama = vize * 0.30 + final * 0.50 + proje * 0.20;
if(ortalama >60)
{
    Console.WriteLine("Geçti ve ortalama= {0}",ortalama);
    //Console.WriteLine("Geçti ve notu"+ortalama);
    // Console.WriteLine($"Geçti ve notu{ortalama}"); //C# 6.0 beraber
}
else
{
    Console.WriteLine("Bütünleme sınavına hazırlan. ortalama={0}",ortalama);
}
```



if-else if-else Örnek 4

```
//ödev ortalamayı harf notuna dönüştürüp ekrana yazdıran program  
  
/*  
    I  
    * 100- 85 AA  
    * 84-70 BB  
    * 69-55 CC  
    * 54-40 DD  
    * 39-0  FF  
    * */
```




switch-case YAPISI

switch (kontrol edilecek değer)

{

case durum1:

işlemler1; // durum1 sağlandığında çalışacak kodlar
break;

case durum2:

işlemler2; // durum2 sağlandığında çalışacak kodlar
break;

default; //Yukarıdakilerin hiçbiri sağlanmıyorsa
break;

}



switch-case YAPISI

```
byte gundegeri=4;  
switch (gundegeri)  
{  
    case 1:  
    case 2:  
    case 3:  
    case 4:  
    case 5: Console.WriteLine("Hafta içi"); break;  
    case 6:  
    case 7: Console.WriteLine("Hafta Sonu");break;  
    default; Console.WriteLine(" Yanlış gün değeri");break;  
}
```



switch-case YAPISI

```
Console.WriteLine("Yapmak istediğiniz işlem nedir ? ");  
Console.WriteLine("1-Şifreleme");  
Console.WriteLine("2-Şifre Çözme");  
c = Console.ReadKey();  
switch (c)  
{  
    case '1': Console.Write("Şifrelenecek Metni Giriniz : "); break;  
    case '2': Console.Write("Şifrelenmiş Metni Giriniz : "); break;  
    default:  
        break;  
}
```



switch-case Örnek 1



```
//öğrenci notu 1- 5 aralığında olacak
/*
 * 1 Zayıf
 * 2 Sorumlu
 * 3 Orta
 * 4 İyi
 * 5 Pekiyi
 * Geçersiz not.
 */
byte ogrnotu;
Console.Write("Öğrenci Notunu Giriniz:");
ogrnotu=Convert.ToByte(Console.ReadLine());
switch (ogrnotu)
{
    case 1:Console.Write("Zayıf");break;
    case 2:Console.Write("Sorumlu");break;
    case 3:Console.Write("Orta");break;
    case 4:Console.Write("İyi");break;
    case 5:Console.Write("Pekiyi");break;
    default:Console.WriteLine("Geçersiz not."); break;
}
```

```
E:\MCSD\MCS D 8 Ocak\5.Hafta\Uygulamalar\Hafta 8\Uygulama 1\Program.cs
Öğrenci Notunu Giriniz:3
Orta
```

```
E:\MCSD\MCS D 8 Ocak\5.Hafta\Uygulamalar\Hafta 8\Uygulama 1\Program.cs
Öğrenci Notunu Giriniz:5
Pekiyi
```

```
E:\MCSD\MCS D 8 Ocak\5.Hafta\Uygulamalar\Hafta 8\Uygulama 1\Program.cs
Öğrenci Notunu Giriniz:8
Geçersiz not.
```

switch-case Örnek 1

```
//Console ekranı rengi değiştirme
string renk;
Console.Write("Console rengi yazınız: mavi, yeşil, kırmızı, sarı: ");
renk = Console.ReadLine();
switch (renk)
{
    case "mavi": Console.BackgroundColor = ConsoleColor.Blue; break;
    case "yeşil": Console.BackgroundColor = ConsoleColor.Green; break;
    case "kırmızı": Console.BackgroundColor = ConsoleColor.Red; break;
    case "sarı": Console.BackgroundColor = ConsoleColor.Yellow; break;
    default: Console.BackgroundColor = ConsoleColor.Black; break;
}
Console.WriteLine("*****");
Console.WriteLine("** renk değiştirme **");
Console.WriteLine("*****");
```

```
E:\MCSD\MCSO 8 Ocak\5.Hafta\Uygulamalar\Hafta5\SwitchCaseOrnek_2\bin\Debug\net6.0\
Console rengi yazınız: mavi, yeşil, kırmızı, sarı: yeşil
*****
** renk değiştirme **
*****
```

```
E:\MCSD\MCSO 8 Ocak\5.Hafta\Uygulamalar\Hafta5\SwitchCaseOrnek_2\bin\Debug\net6.0\
Console rengi yazınız: mavi, yeşil, kırmızı, sarı: sarı
*****
** renk değiştirme **
*****
```

```
E:\MCSD\MCSO 8 Ocak\5.Hafta\Uygulamalar\Hafta5\SwitchCaseOrnek_2\bin\Debug\net6.0\
Console rengi yazınız: mavi, yeşil, kırmızı, sarı: mavi
*****
** renk değiştirme **
*****
```



switch-case YAPISI

Switch-case yapısında aşağıdaki türler kullanılabilir.

sbyte,
byte,
short,
ushort,
int,
uint,
long,
ulong

char,
string