### IF ELSE KARAR YAPISI

if else deyimi sayesinde belli bir koşul sağlandığında söz konusu komutlar çalıştırılır, o belli koşullar sağlanmadığında çalıştırılmaz ya da başka komutlar çalıştırılır. Kullanılışı şu şekildedir:

|  |  |
| --- | --- |
| Tek bir komut var ise | Birden fazla komut var ise |
| if(koşul) komut1;  veya  if(koşul)  komut1; | if(koşul)  {  komutlar;  ……  komutlar;  } |
| if(koşul)  komut1;  else  komut2; | if(koşul)  {  komutlar;  ……  komutlar;  }  else  {  komutlar;  ……  komutlar;  } |
| if(koşul)  komut1;  else if (koşul)  komut2;  else  komut3; | if(koşul)  {  komutlar;  ……  komutlar;  }  else if (koşul)  {  komutlar;  ……  komutlar;  }  else  {  komutlar;  ……  komutlar;  } |

Yukarıdaki örneklerde eğer koşul sağlanırsa 1. komutlar, sağlanmazsa 2. komutlar çalıştırılır. if veya else'in altında birden fazla komut varsa bu komutları parantez içine almak gerekir. if veya else'in altında tek komut varsa bu komutları parantez içine almak gerekmez.

Örnek 1:

using System;

class AkisKontrolMekanizmalari

{

static void Main()

{

int a = 9, b = 7;

if (a < b)

Console.Write("a b'den küçük");

else

Console.Write("a b'den küçük değil");

Console.ReadLine();

}

}

Örnek 2:

using System;

class AkisKontrolMekanizmalari

{

static void Main()

{

int a = 5, b = 7;

if (a < b)

{

Console.WriteLine("a b'den küçük");

Console.Write(a);

}

else

{

Console.WriteLine("a b'den küçük değil");

Console.Write(b);

}

Console.ReadLine();

}

}

* if else yapılarında else kısmının bulunması zorunlu değildir. Bu durumda sadece koşul sağlandığında birşeyler yapılacak, koşul sağlanmadığında bir şeyler yapılmayacaktır.

Örnek 3:

using System;

class AkisKontrolMekanizmalari

{

static void Main()

{

int a = 5, b = 7;

if (a < b)

Console.WriteLine("a b'den küçük");

Console.ReadLine();

}

}

* if else bloklarının aşağıdaki gibi kullanımı da mümkündür:

Örnek 4:

using System;

class AkisKontrolMekanizmalari

{

static void Main()

{

Console.Write("Cinsiyetinizi girin (e, k): ");

char cins = Convert.ToChar(Console.ReadLine());

if (cins == 'e')

Console.Write("cinsiyet : Erkek");

else if (cins == 'k')

Console.Write("cinsiyet : kız");

else

Console.Write("Lütfen cinsiyetinizi doğru giriniz!");

Console.ReadLine();

}

}

Bu program kullanıcıdan cinsiyetini girmesi istemekte, eğer kullanıcının girdiği harf e ise ekrana cinsiyet : Erkek yazmakta, eğer girdiği harf e değilse ise bu sefer kullanıcının girdiği harfi k mı değil mi diye incelemekte, eğer k girmişse ekrana cinsiyet: Kız yazmakta, bunların dışında bir harf girdiğinde de ekrana Lütfen cinsiyetinizi doğru giriniz! yazmaktadır. Bu şekilde bu bloklar daha da uzatılabilir. Yani else if satırının bir tane olması zorunlu değildir. Ancak tabii ki else satırının yalnızca bir tane olması gerekir.

BİRDEN FAZLA İF ELSE İF KULLANIMI.

using System;

class AkisKontrolMekanizmalari

{

static void Main()

{

Console.Write("Cinsiyetinizi girin (e, k): ");

char cins = Convert.ToChar(Console.ReadLine());

if (cins == 'e')

Console.Write("cinsiyet : Erkek");

else if (cins == 'k')

Console.Write("cinsiyet : kız");

else if (cins == 'E')

Console.Write("cinsiyet : Erkek");

else if (cins == 'K')

Console.Write("cinsiyet : kız");

else

Console.Write("Lütfen cinsiyetinizi doğru giriniz!");

Console.ReadLine();

}

}

**C# iç içe if else kullanılmasına izin verir:**

if(koşul1)

{

if(koşul2)

komut1;

else

{

komut2;

komut3;

}

}

else

komut4;

Örnek 5:

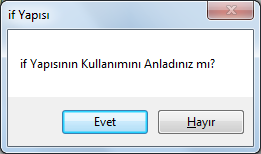
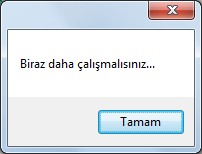
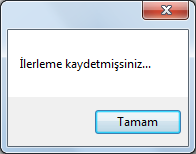
Örnek 6:

Burada kullanıcıya MessageBox ile bir soru soruyoruz.

Mesaj penceresi Evet ve Hayır butonları olan bir diyalog kutusu şeklindedir.

Kullanıcı eğer(if) Evet butonunu tıklarsa ilk komutu değilse(else) ikinci komutu çalıştıracaktır.

MessageBox metodunda MessageBoxButtons.YesNo no ile Evet ve Hayır butonlarını ekledik. DialogResult.Yes komutu ile de cevabın Evet olup olmadığını kontrol ettik.

using System;

using System.Collections.Generic;

using System.ComponentModel;

using System.Data;

using System.Drawing;

using System.Linq;

using System.Text;

using System.Windows.Forms;

namespace WindowsFormsApplication1

{

public partial class Form1 : Form

{

public Form1()

{

InitializeComponent();

}

private void Form1\_Load(object sender, EventArgs e)

{

// Burada kullanıcıya MessageBox ile bir soru soruyoruz.

// Mesaj penceresi Evet ve Hayır butonları olan bir diyalog kutusu şeklindedir.

// Kullanıcı eğer(if) Evet butonunu tıklarsa ilk komutu değilse(else) ikinci komutu çalıştıracaktır.

if (MessageBox.Show("if Yapısının Kullanımını Anladınız mı?", "if Yapısı", MessageBoxButtons.YesNo) == DialogResult.Yes)

MessageBox.Show("İlerleme kaydetmişsiniz..." );

else

MessageBox.Show("Biraz daha çalışmalısınız...") ;

// MessageBox metodunda MessageBoxButtons.YesNo no ile Evet ve Hayır butonlarını ekledik. DialogResult.Yes komutu ile de cevabın Evet olup olmadığını kontrol ettik.

}

}

}

Örnek 7:

Bu örnekde de 5 ile 3'ü çarptık sonuç 15 çıkıyorsa uyarı olarak veriyoruz.

Değilsede işlem sonucunun eşit olmadığını gösteriyoruz.

using System;

using System.Collections.Generic;

using System.ComponentModel;

using System.Data;

using System.Drawing;

using System.Linq;

using System.Text;

using System.Windows.Forms;

namespace WindowsFormsApplication1

{

public partial class Form1 : Form

{

public Form1()

{

InitializeComponent();

}

private void Form1\_Load(object sender, EventArgs e)

{

// Bu örnekde de 5 ile 3'ü çarptık sonuç 15 çıkıyorsa uyarı olarak veriyoruz.

// Değilsede işlem sonucunun eşit olmadığını gösteriyoruz.

if ((5 \* 3) == 15)

MessageBox.Show("Doğru bir işlem...") ;

else

MessageBox.Show("İşlem sonucu 15 değil...") ;

}

}

}

### switch

switch deyimi bazı if else deyimlerinin yaptığı işi daha az kodla yapar. Genellikle bazı karmaşık if else bloklarını kurmaktansa switch'i kullanmak programın anlaşılırlığını artırır. Ancak tabii ki basit if else bloklarında bu komutun kullanılması gereksizdir. Kullanımı:

switch(ifade)

{

case sabit1:

komut1;

break;

case sabit2:

komut2;

break;

default:

komut3;

break;

}

Bu switch deyimiyle ilgili bilmeniz gerekenler:

* İfadenin ürettiği değer hangi case sabitinde varsa o "case"deki komutlar işletilir. Eğer ifadenin ürettiği değer hiçbir case sabitinde yoksa default casedeki komutlar işletilir.

Örnek 1:

using System;

using System.Collections.Generic;

using System.Linq;

using System.Text;

namespace ConsoleApplication1

{

class Program

{

static void Main(string[] args)

{

int a = 4;

switch (a)

{

case 3:

Console.Write("deneme1");

break;

case 4:

Console.Write("deneme2");

break;

case 5:

Console.Write("deneme3");

break;

default:

Console.Write("deneme4");

break;

}

Console.ReadLine();

}

}

}

Örnek 2:

using System;

using System.Collections.Generic;

using System.Linq;

using System.Text;

namespace ConsoleApplication1

{

class Program

{

static void Main(string[] args)

{

Console.WriteLine("4 e basıp isim girebilirsiniz");

int a =Convert .ToInt16 ( Console.ReadLine());

switch (a)

{

case 4:

string b = Console.ReadLine();

Console.Write(b);

break;

default:

Console.Write("4 girmediniz; çıkış için tuşa basın");

Console.ReadKey();

break;

}

}

}

}

* Eğer programımızın bir casedeyken farklı bir casee gitmesini istiyorsak goto anahtar sözcüğü kullanılır.

Örnek 3:

using System;

using System.Collections.Generic;

using System.Linq;

using System.Text;

namespace ConsoleApplication1

{

class Program

{

static void Main(string[] args)

{

int a = 5;

switch (a)

{

case 4:

Console.Write("Burası case 4'tür.");

break;

case 5:

Console.WriteLine("Şimdi case 4'e gideceksiniz.");

goto case 4;

default:

Console.Write("deneme4");

break;

}

Console.ReadLine();

}

}

}

* goto satırı kullanılmışsa break; satırının kullanılmasına gerek yoktur.
* Eğer farklı case sabitlerinin aynı komutları çalıştırmasını istiyorsak şöyle bir program yazılabilir:

Örnek 4:

Bu programda a değişkeni 4 olsa da 5 olsa da aynı komutlar çalıştırılacaktır.

using System;

using System.Collections.Generic;

using System.Linq;

using System.Text;

namespace ConsoleApplication1

{

class Program

{

static void Main(string[] args)

{

int a = 4;

switch (a)

{

case 4:

case 5:

Console.Write("5 seçildi");

break;

default:

Console.Write("4 yada 5 değildir");

break;

}

Console.ReadLine();

}

}

}

* case anahtar sözcüğünün yanındaki ifade mutlaka ya sabit ya da sabitlerden oluşan bir ifade olmalıdır.
* default durumunu istediğiniz yere yazabilirsiniz, aynı şekilde istediğiniz casei de istediğiniz yere yazabilirsiniz. Yani caselerin sırası önemli değildir.
* Bir switch bloğunda default durumu bulunmak zorunda değildir.
* switch'in parantez içindeki ifadesi bir değişken olabileceği gibi, bir sabit ya da ifade de olabilir.