

9- PROGRAM YAPISAL KOMUTLARI DÖNGÜLER (LOOPS)

`while()`, `do-while()`, `for()`, `foreach()`

Döngü (loop) : Bir koşul sağlandığı (true olduğu) sürece bir veya birden çok komutun tekrarlanması işlemidir.

C# Döngüler (loops)

- ❑ **While ()**
- ❑ **do-while ()**
- ❑ **for ()**
- ❑ **foreach ()**

```
while ( koşul )  
{  
    işlemler;  
}
```

```
do {  
    işlemler;  
} while (koşul );
```

```
for (başlangıç ; koşul ; arttırma)  
işlemler;
```

```
foreach ( tip değişken in dizi )  
işlemler;
```

9- PROGRAM YAPISAL KOMUTLARI

9.1 - while () döngüsü

```
while ( koşul )  
{  
    -----;  
    -----;  
    -----;  
}
```

} Döngü Bloğu

koşul **true** olduğu sürece *döngü bloğu* içerisindeki komutlar çalışacaktır.

9- PROGRAM YAPISAL KOMUTLARI

while () döngüsü

```
int x=5;
while ( x<5 )
{
    Console.Write(x);
}
```

//koşul sağlanmadığından döngüye girilmez

```
int x=5;
while ( x<=5 )
{
    Console.Write(x);
}
```

//koşul sürekli sağlanıyor, sonsuz döngü

```
int x=5;
while ( x>=1 )
{
    Console.Write(x);
    x--;
}
```

// 54321 yazar

NOT : Bu tür döngülerde *döngü bloğu içerisinde döngünün koşulunu etkileyen mutlaka bir komut satırı* olmalıdır yada döngüden **break** veya **return** ile çıkılmalıdır. Aksi halde sonsuz döngü olacaktır.

- ☐ Koşul sağlanmaz ise döngüye hiç girilmez.
- ☐ Koşul sürekli sağlanıyor ise sonsuz döngü oluşur.

9- PROGRAM YAPISAL KOMUTLARI

- ❑ **break** : İçinde bulunduğu döngüden çıkar.
- ❑ **continue** : İçinde bulunduğu döngünün *koşul* ifadesine gönderir.
- ❑ **return** : İçinde bulunduğu fonksiyondan çıkar. Eğer bu fonksiyon Main() fonksiyon ise program sonlanır. Return ile değer de gönderebiliriz. Bakınız Fonksiyonlar

9- PROGRAM YAPISAL KOMUTLARI

while () döngüsü

```
int x=0;
while (x<10)
{
    x++;
    if( (x%2) == 0 ) continue;
    Console.Write(x);
}
```

/* 1 3 5 7 9 yazar
continue ile döngünün başına (koşul ifadesine)
gönderir
*/

```
int x=0;
while (x<10)
{
    x++;
    if( (x%2) != 0 ) break;
    Console.Write(x);
}
```

// hiçbir şey yazmaz
/* break döngünün dışına çıkılabilir */

while () döngüsü

```
int x=0;
while ( x<10 )
{
    x++;
    if( (X%2)== 0 )
        Console.Write(x);

}
```

//2 4 6 8 10

```
int x=1;
while ( x<=5 )
{
    int y= Math.Pow(2,x);
    Console.Write(" 2 üssü {0} ={1}" , x, y );
    x++;
}
```

//2 4 8 16 32

9- PROGRAM YAPISAL KOMUTLARI

while () döngüsü

```
int x=6;
while ( x !=5 )
{
    int y = x * x ;
    Console.Write( y );
    x--;
}
Console.Write( x );
Console.Write( y ); // Hata verir döngü içerisinde tanımlıdır. Döngü bitince y bellekten atılır
```

NOT: Döngünün içerisinde tanımlanan değişkenler sadece döngü içerisinde geçerlidir. Döngünün dışına çıkıldığında değişken bellekten atılacaktır.

Örneğimizdeki **y** değişkeni döngü içinde tanımlandığından, sadece döngü içinde geçerlidir.

do - while()

9- PROGRAM YAPISAL KOMUTLARI

9.2 - **do-while ()** döngüsü

```
do  
{  
    -----;  
    -----;  
    -----;  
} while ( koşul ) ;
```

} Döngü Bloğu

while döngüsünden farklı olarak koşul sondadır. Dolayısıyla şart hiç sağlamasa da döngü *en az bir kez* çalışacaktır.

9- PROGRAM YAPISAL KOMUTLARI

do-while () döngüsü

```
int x=5;  
while (x<5)  
{  
    Console.Write("A");  
}
```

```
Console.Write("B");  
// çıktı olarak B
```

```
int x=5;  
do  
{  
    Console.Write("A");  
} while (x<5);
```

```
Console.Write("B");  
// çıktı olarak AB
```

do-while () döngüsü

```
int x=0;
while (x<10)
{
    x++ ;
    if( (x%2)==0 ) continue;
    Console.Write(x);
}
```

/* **continue** ile döngünün başına (koşul ifadesine) gönderir */

```
int x=0;
do
{
    x++;
    if( (x%2)==0 ) continue;
    Console.Write(x);
} while (x<10);
```

/* **continue** ile döngünün sonuna (koşul ifadesine) gönderir */

```
for ( ; ; )  
{  
    ...  
}
```

9- PROGRAM YAPISAL KOMUTLARI

9.3- for () döngüsü

Döngüye **ilk girildiği zaman** yapılacak işlemler. Bir kez çalışır. Döngüye ilk girildiğinde

Döngünü **koşulu**. Koşulun sonucu **true** olduğu sürece döngü dönmeye devam eder. Döngü her döndüğünde buraya bakılır koşul **true** ise devam edilir

Döngü her döndüğünde önce buraya bakılır, bu bölümde bir işlem varsa yapılır. Genellikle döngünün **artırma** veya **azaltma** işlemi burada yapılır.

```
for ( I ... ; II ... ; III ... )  
{  
    ..... ;  
    ..... ;  
    ..... ;  
}
```

Döngü bloğu; döngü her döndüğünde yapılacak işlemler.

for () döngüsü

```
int x=0;
while (x<5)
{
    Console.Write(x);
    x++;
}

// 0 1 2 3 4
```

```
for (int x=0; x<5 ; x++)
{
    Console.Write(x);
}

// 0 1 2 3 4
```


for () döngüsü

```
for (int x=0 ; x<50 ; x+=10 )  
{  
    Console.Write (x);  
}
```

<u>x</u>
0
10
20
30
40

9- PROGRAM YAPISAL KOMUTLARI

for () döngüsü

// ekrana ne yazar ?

```
for (int x=50 ; x<50 ; x+=10 )  
{  
    Console.Write (x);  
}
```

9- PROGRAM YAPISAL KOMUTLARI

for () döngüsü

```
for (int x=1 , y= 5 ; x<y ; x++ , y--)  
{  
    Console.Write ( " {0} {1} {2} \n " , x , y , x * y );  
}  
Console.Write ( x * y ); // Bu satır HATALI neden ?
```

<u>x</u>	<u>y</u>	<u>x*y</u>
1	5	5
2	4	8

9- PROGRAM YAPISAL KOMUTLARI

Örnek

// Ekrana ne yazar ?

```
for ( int x =5 , y= 5 ;  x!=y ;  x++, y-- )  
{  
    Console.WriteLine(x*y );  
}  
Console.WriteLine( "B" );
```

B yazar

for () döngüsü

```
int x=1 ;  
for (    ; x<5 ; x++ ) Console.Write ( x );
```

```
int x=1 ;  
for (    ; x<5 ;    )  
{  
    Console.Write ( x );  
    x++;  
}
```

```
int x=1 ;  
for (    ; ;    )  
{  
    if( x>=5 ) break;  
    Console.Write ( x );  
    x++;  
}
```

for () döngüsü

```
for (int x = 1 ; x>0 && x<3 ; x++ )  
{  
    Console.Write ( x );  
    x++;  
}  
// 1
```

9- PROGRAM YAPISAL KOMUTLARI

Örn: Rastgele tutulan sayılardan üst üste aynı sayı gelene kadar devam ettiren ve kaçınıcı denemede bulunduğunu yazdıran program

```
int say1, say2 ;
Random r = new Random();
int n=0 ;
for ( ; ; )
{
    n++;
    say1 = r.Next (0,100) ; // [0-100 )
    say2 = r.Next ( ) %100 ; // [0-100 )
    if ( say1 == say2 ) break ;
}
Console.Write (“{0} kez denendi “, n );
```

foreach ()

9- PROGRAM YAPISAL KOMUTLARI

9.4 - foreach() döngüsü

Bir grup elemanı ilk elemandan başlayıp son elamana kadar teker teker ele alan bir döngüdür.

```
foreach ( tip  degisken in koleksiyon)
{
    ....
}
```

İlk elemandan başlamak üzere her döndüğünde *grup* ' taki sırası gelen elemanı alıp *deger* ' e aktarır. *tip , deger* ' deki verinin tipidir.

9- PROGRAM YAPISAL KOMUTLARI

9.4 - foreach() döngüsü örneği

```
string s= "Dün hava kar yağışlıydı." ;  
foreach ( string kelime in s.Split(' '))  
{  
    Console.Write ("\\n {0} " , kelime );  
}
```

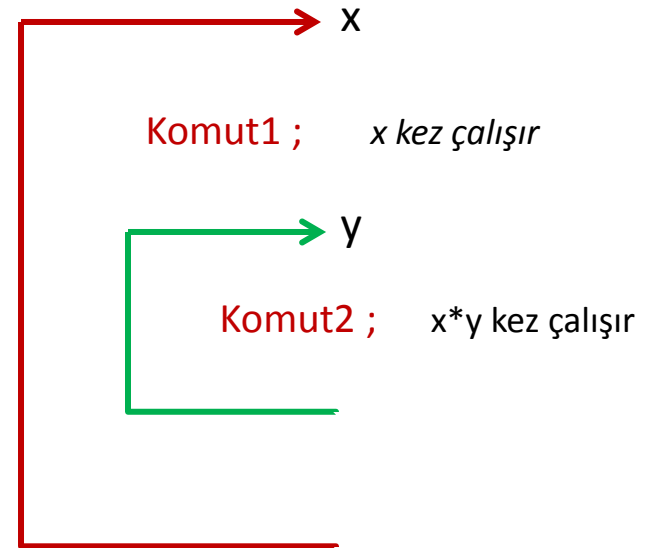
Ekran Çıktısı :

Dün
hava
kar
yağışlıydı.

9- PROGRAM YAPISAL KOMUTLARI

9.5 - iç-içe döngüler (nested loops)

```
for ( int x=1 ; x<=5 ; x++ )  
{  
    Komut1;           // 5 kez çalışır  
  
    for (int y=1 ; y<=3 ; y++ )  
    {  
        Komut2;       // 5*3 kez çalışır  
    }  
}
```



İçteki döngü (y), dıştaki döngünün(x) bir komutu gibidir. Dolayısıyla dıştaki döngü her döndüğünde içteki döngü sonuna kadar dönecektir.

9- PROGRAM YAPISAL KOMUTLARI

9.5 - iç içe döngüler

```
for ( int x=1 ; x<=3 ; x++ )
{
    Console.WriteLine( x ); // bu satır x döngüsü kadar çalışır 3 kez

    for (int y=1 ; y<=4; y++ )
    {
        Console.Write( y ); // bu satır x in her değeri için, y döngüsü kadar çalışır. Yani x* y kez çalışır
    }
}
```

x: 1
y: 1 2 3 4

x: 2
y: 1 2 3 4

x: 3
y: 1 2 3 4

x, 3 kez çalışırken ,
y değeri x'in her değeri için 4 kez çalışacaktır.

Sonuç olarak en içteki döngüdeki bir satır
3*4=12 kez çalışmış olur.

9- PROGRAM YAPISAL KOMUTLARI

9.5 - iç içe döngüler. Örnek;

```
for ( int x =1 ; x<=4 ; x++ )  
{  
    for (int y=1 ; y<=3 ; y++ )  
    {  
        Console.Write( y );  
    }  
    Console.WriteLine( ); // satırbaşı yapar  
}
```

```
1 2 3  
1 2 3  
1 2 3  
1 2 3
```

```
1 1 1  
2 2 2  
3 3 3  
4 4 4
```

Şeklinde çıktı vermesi için programda nasıl bir değişiklik yapılmalıdır.

???

Çeşitli Örnekler

9- PROGRAM YAPISAL KOMUTLARI

Örnek-5: Klavyeden girilen bir sayının faktöriyelini bulup yazdıran program

```
int f=1;  
int x = int.Parse( Console.ReadLine () );  
for ( int y=1 ; y<=x ; y++ ) f = f * y ;  
Console.Write ( " {0} != {1} ", x , y );
```

Örneğin x=4 için ekrana;

4 !=24

yazar

Örnek-1: Sayı tahmin oyunu

```
static void Main ( string [ ] args)
{
    string cev = "E";
    while ( cev.ToUpper() == "E" )
    {
        int sayac = 1;
        Random r = new Random();
        int x = r.Next (1,101); // [1,101) arasında rastgele sayı türetir
        Console.Clear();
        Console.Write("Bir sayı tahmin ediniz");
        while ( true )
        {
            int tahmin = int.Parse( Console.ReadLine() );
            sayac++;
            if (tahmin > x) { Console.Write("AŞAĞI");}
            if (tahmin < x) { Console.Write("YUKARI"); }
            if (tahmin == x)
            {
                Console.Write("TEBRİKLER {0}. tahminde buldunuz\n\n", sayac);
                Console.Write("Devam edilecek mi [e/h]");
                cev = Console.ReadLine();
                break; // 2. while döngüsünden çıkar
            }
        }
    }
} // Main fonksiyonu sonu
```

9- PROGRAM YAPISAL KOMUTLARI

Örnek -2 : ASCII Tablosunu ekrana 10' arlı sütunlar halinde yazdırınız ?

```
using System;

namespace ornek_5_3_2
{
    class Program
    {
        static void Main(string[] args)
        {
            for (int m = 0 ; m < 256 ; m++ )
            {
                if ( ( m % 10) == 0 ) Console.WriteLine( ) ; // Satır başı
                Console.Write(" {0} -> {1} ", m , (Char) m );
            }
        }
    }
}
```

...									
65->A	66->B	67->C	68->D	69->E	70->F	71->G	72->H	73->I	74->J
75->K	76->L	77->M	78->N	79->O	80->R	81->S	82->T	83->U	84->V
....									

Örnek -3 : 1 den başlayıp klavyeden girilen sayı kadar aşağıdaki şekilde görüldüğü gibi (15 için) ‘* ‘ yazdıran programı yazınız.

```

      *
     ***
    *****
   *********
  ***********
 *****
*****
*****
*****
*****

```

//Cevap-3;

static void Main (string[] args)

{

int n;

do

{

Console.Write ("Please enter the number of rows ([1, 15]: ");

n = Int32.Parse(Console.ReadLine());

} while (n < 1 || n > 15);

for (int row = 1; row <= n; row ++)

{

for (int space = 0; space < n - row ; space ++) Console.Write(' ');

for (int col = 0 ; col < 2 * row -1 ; col ++) Console.Write('*') ;

Console.WriteLine();

}

}

Örnek -4 : 1 den başlayıp klavyeden girilen sayı kadar aşağıdaki şekilde görüldüğü gibi sağa dayalı (9 için) '*' yazdıran programı yazınız.

```
      *
     **
    ***
   ****
  *****
 *****
*****
*****
*****
```

// Cevap-4;

```
static void Main ( string[] args )  
{  
    int n=int.Parse ( Console.ReadLine() );  
    for (int i=1; i<=n ; i++)  
    {  
        Console.WriteLine();  
        for (int j=9 ; j>=i ; j-- ) Console.Write(" ") ;  
        for (int k=1 ; k<=i ; k++) Console.Write("*");  
    }  
  
} // end of method Main
```

9- PROGRAM YAPISAL KOMUTLARI

Örnek-6 : Klavyeden girilen 10 luk sistemin 2 lik karşılığını bulan program

```
string s="";
int x=0 , say=0;
x = Convert.ToInt32(Console.ReadLine() );
if( x>0)
{
    do
    {
        int kalan = (x%2) ;
        s += kalan.ToString() ;
        x /=2 ;;
    } while (x>0);
}

int boyu = s.Length;
while ( boyu>0)
{
    Console.Write ( " {0} " , s.Substring ( (boyu-1) , 1 );
    boyu --;
}
```

15 -> 1 1 1 1
16 -> 1 0 0 0