

## CONSOLE EKRANI-6

### CONSOLE EKRANI

- ✓ *Diziler*

### KAZANIMLAR

- ✓ *Çok boyutlu ve tek boyutlu dizilerinin mantığını öğrenir.*
- ✓ *Foreach döngüsünü öğrenir.*
- ✓ *Dizilerle ilgili problemleri çözer.*

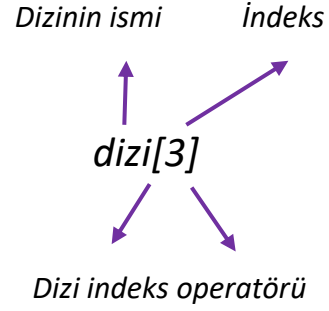


## Diziler

Bellekte ard arda yer alan aynı türden nesneler kümesine dizi (array) denilir. Bir dizi içerisindeki bütün elemanlara aynı isimle ulaşılır. Yani dizideki bütün elemanların isimleri ortaktır. Elemanlar arasındaki ayırt edici özellik, bellekteki yeridir.

5 elemanlı bir tamsayı dizisi bellekte aşağıdaki gibi yerleştirilir

dizi[]	
1020	dizi[0]
-20	dizi[1]
5	dizi[2]
0	dizi[3]
16587	dizi[4]



C# dilinde bütün diziler **System.Array** sınıfından türetilir. Bir dizinin oluşturulması iki farklı şekilde yapılabilir. Birinci yöntem bildirim ve tanımlamanın aynı satırda yapılması;

```
string [] kişiler = new string[20]
```

şeklinde. Burada 20 elemanlı kişiler isminde string türünden bir dizi tanımlanmıştır. İkinci yöntem olarak diziyi aşağıdaki şekilde bildirim ve tanımlama ayrı satırda olacak şekilde oluşturmak istersek;

```
string [] kişiler;
```

```
kişiler = new string[20];
```

oluşturabiliriz. Bu 2 yöntem dışında **dizi** oluşturulurken içine değer girmek için şu yöntem kullanılabilir.

```
String [] kişiler = {"Ahmet", "Mehmet", "Ayşe", "Fatma", "Canan"}
```

Burada da yine **kişiler** isminde bir **dizi** oluşturularak aynı satırda diziyi değer girilmesi sağlanmıştır. Yukarıdaki dizide "Mehmet" isimli elemana erişmek için;

```
String isim = kişiler[1];
```

Son elemanına erişmek için;

```
String isim = kişiler[4];
```

İlk elemanına erişmek için ise;

```
String isim = kişiler[0];
```



**Örnek1:** Aşağıdaki kod parçasını bilgisayarımızda deneyelim..

```
string[] gunler = { "Pazartesi", "Salı", "Çarşamba", "Perşembe", "Cuma", "Cumartesi", "Pazar" };

for (int i = 0; i < gunler.Length; i++)
{
    Console.WriteLine(gunler[i]);
}
```

**Örnek2:** elemanlı isimler isminde bir dizi oluşturarak ekranda foreach döngüsü kullanarak yazdıralım.

```
Static void Main( string [] args)
{
    String [] kişiler={ "Ahmet","Ali","Veli","Ayşe","Fatma","Canan"}
    Foreach(string isim in kişiler)
    {
        Console.WriteLine(isim);
    }
    Console.ReadKey();
}
```



#### **Dikkat Foreach Kullanımı**

For ile aynı görevi vardır.Tek farkı boyutu belli olmayan işlemlerde for'u kullanamıyorduk fakat bunu kullanabiliyoruz. Yani biz **Foreach'i** listeler ya da diziler üzerinde işlem yapmak için kullanılan **döngü** yapısıdır. Özellikle dizilerde kullanıyoruz.



**Örnek3:** Rastgele üretilen 5 adet sayıyı büyükten küçüğe doğru alt alta gösterin.

```
using System;

namespace ConsoleApp1
{
    class Program
    {
        static void Main(string[] args)
        {
            int[] dizi = new int[5];
            Random rastgele = new Random();

            for (int i = 0; i < dizi.Length; i++)
            {
                dizi[i] = rastgele.Next(0, 100);
            }
            Array.Sort(dizi);
            Array.Reverse(dizi);
            for (int i = 0; i < dizi.Length; i++)
            {
                Console.WriteLine(dizi[i]);
            }
        }
    }
}
```



**Dikkat:** Yukardaki örnekte dizi metodlarından “Array.Sort” ve “Array.Reverse” yeralıyor. Diğer dizi içerisindeki metotlara değinelim.

**Contains:** Bu metod dizi içinde elemanın var olup olmadığını kontrol eder ve geriye **boolean** türünde yani **true** ya da **false** olarak bir değer döndürür.

**IndexOf:** Contains metodunda yapmış olduğumuz aramada aranan değer listede var olup olmadığını öğrenmiş olduk. **IndexOf** metodu ile yapacağımız aramada ise aranan değer **index** numarasını bize gönderecektir. Yani aradığımız değer bu dizinin kaçınıcı değeri olduğunu öğrenebileceğiz.

```
int sıra=Array.IndexOf(aranacak dizi,aranacak ifade);
```

```
indexNo = liste.IndexOf(arananDeger);
```



**Örnek4:** Kullanıcının girmiş olduğu sayının dizide olup olmadığını bulan, varsa kaç adet bulunduğunu ekranda gösteren dizi örneği:

```
using System;

namespace ConsoleApp1
{
    class Program
    {
        static void Main(string[] args)
        {
            {
                int[] sayilar = { 60, 80, 55, 42, 13, 24, 26, 25, 26, 55, 45, 50, 50, 40};
                int sayi, kontrol = 0;
                Console.Write("Bir sayı giriniz :");
                sayi = Convert.ToInt32(Console.ReadLine());
                for (int i = 0; i < sayilar.Length; i++)
                {
                    if (sayi == sayilar[i])
                    {
                        kontrol++;
                    }
                }
                if (kontrol == 0)
                {
                    Console.WriteLine("Sayı dizi içinde bulunamadı...");
                }
                else
                {
                    Console.WriteLine("Bu sayı dizide {0} adet var", kontrol);
                }
                Console.ReadLine();
            }
        }
    }
}
```



**Örnek5:** Kullanıcı tarafından dizinin boyutunu belirledikten sonra diziye veri atmayı veren program adımları...

```
using System;
namespace ConsoleApp4
{
    class Program
    {
        static void Main(string[] args)
        {
            int sayi = 0; ;
            Console.WriteLine("Sınıf mevcudunuz nedir?");
            sayi = Convert.ToInt32(Console.ReadLine());
            string[] ogrenci = new string[sayi];

            for (int i = 0; i < sayi; i++)
            {
                Console.Write(" {0}.Öğrenci Adını Girin:", i + 1);
                ogrenci[i] = Console.ReadLine();
            }
            Console.WriteLine("sisteme verileriniz yüklenmiştir...");
        }
    }
}
```

**Örnek6:** Kullanıcıdan alınan metnin içinde bulunan sesli harf sayısını bulan c# console uygulaması

```
static void Main(string[] args)
{
    char[] harfler = { 'a', 'e', 'ı', 'i', 'o', 'ö', 'u', 'ü' };
    string metin;
    int sayac = 0;
    Console.WriteLine("Metin giriniz : ");
    metin = Console.ReadLine();
    for (int i = 0; i < harfler.Length; i++)
    {
        for (int j = 0; j < metin.Length; j++)
        {
            if (metin[j] == harfler[i])
            {
                sayac++;
            }
        }
    }
    Console.WriteLine(sayac);
    Console.ReadLine();
}
```



## ÇOK BOYUTLU DİZİLER

```
int[] BirinciSınavNotları = new int[10];
```

BirinciSınavNotları[5] = 65

```
int[,] TümSınavNotları = new int[10,2];
```

↑ Değer

TümSınavNotları[5,1] = 30



**Örnek7:** 2 boyutlu bir dizinin satır ve sütun sayısının kullanıcı tarafından girilmesi işlemini gerçekleştireceğiz. Kullanıcı 2 boyutlu oluşturacağımız dizinin kaç satır ve kaç sütundan oluşacağını belirleyecek ve oluşturulan bu dizi içerisine rasgele 0-100 arası sayılar gireceğiz.

```
static void Main(string[] args)
{
    Console.Write("Satır sayısını gir>>");
    int a = Convert.ToInt32(Console.ReadLine());
    Console.Write("Sütun sayısını gir>>");
    int b = Convert.ToInt32(Console.ReadLine());
    Console.WriteLine("=====");
    Random rnd = new Random();

    int[,] dizi2 = new int[a, b];
    for (int i = 0; i < a; i++)
    {
        for (int j = 0; j < b; j++)
        {
            dizi2[i, j] = rnd.Next(0, 50);
            Console.Write(dizi2[i, j] + "-");
        }
        Console.WriteLine();
    }
    Console.ReadKey();
}
```





