



Kırklareli Üniversitesi  
Lüleburgaz Meslek Yüksekokulu

# Görsel Programlama-I

## Hafta-11 : String İşlemleri

Öğr. Gör. Özcan EKEN  
ozcan.eken@klu.edu.tr

## String

String'ler, karakterlerin bir araya gelmesi ile oluşturulan kelimeler cümleler topluluğudur. C# dilinde string'ler üzerinde işlem yapmayı kolaylaştıran pek çok metot ve özellik bulunmaktadır. String'ler;

```
string Değişken_Adı = İlk_değer ;
```

```
string adsoyad = "Özcan EKEN";
```

şeklinde tanımlanabileceği gibi, ilk değer ataması yapılamadan da tanımlanabilir.

```
string adsoyad;
```

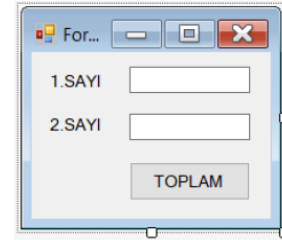
String değişkenler; hafızada 10 byte ( 0 ile 2 milyar karakter arası) yer kaplarlar.

String'ler üzerinde işlem gerçekleştirilirken, **String sınıfı** ya da **StringBuilder sınıfı** kullanılır.

## ToString Metodu

Belirtilen bir değişken türünü string türüne dönüştürmek için kullanılır.

```
private void btn_topla_Click(object sender, EventArgs e)
{
    int sonuc;
    sonuc = Convert.ToInt32(textBox1.Text) + Convert.ToInt32(textBox2.Text);
    MessageBox.Show(sonuc);
}
```



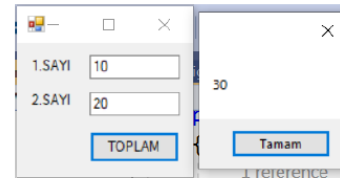
Yukarıdaki kod çalıştırıldığında, sonuc değişkeninin içeriği integer olduğundan dolayı, mesaj penceresi ile görüntülemek istediğimizde string değere dönüştürülmediğinden dolayı



hatası verecektir.

```
MessageBox.Show(sonuc.ToString());
```

yazdığımızda, programımız doğru çalışır.



## Copy Metodu

Bir string' in içeriğini başka bir stringe kopyalamak amacı ile kullanılır.

```
Hedef_String_Adı = String.Copy(Kaynak_String_Adı);
```

```
static void Main(string[] args)
{
    string a,b;
    a = "Bilgisayar";
    b = String.Copy(a);
    Console.WriteLine(b);
    Console.ReadKey();
}
```

Yukarıdaki kod çalıştırıldığında, a değişkenin içeriği b değişkenine kopyalanarak **Bilgisayar** değerini alacaktır.

## Concat Metodu

Birden fazla stringi birleştirerek tek bir string haline getirmek amacıyla kullanılır. Eklenen string diğer stringin sonuna eklenecektir.

**String.Concat(string 1, string 2, ....., string n);**

```
static void Main(string[] args)
{
    string a,b,c;
    a = "Bilgisayar";
    b = "Programcılığı";
    c = String.Concat(a," ",b);
    Console.WriteLine(c);
    Console.ReadKey();
}
```

Yukarıdaki kod çalıştırıldığında, a string değişkeni, boşluk karakteri, ve b string değişkeni içeriği birleştirilerek ekrana **Bilgisayar Programcılığı** yazacaktır.

## Empty Metodu

Bir stringin içeriğini boşaltmak için kullanılır.

**String\_değişken\_adı = String.Empty;**

```
static void Main(string[] args)
{
    string a,b,c;
    a = "Bilgisayar";
    b = "Programcılığı";
    b = String.Empty;
    c = String.Concat(a,b);
    Console.WriteLine(c);
    Console.ReadKey();
}
```

Yukarıdaki kod çalıştırıldığında, b değişkeninin içeriğini Empty metodu ile boşalttığımız için ekrana **Bilgisayar** yazacaktır.

## Split Metodu

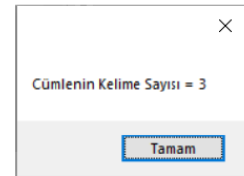
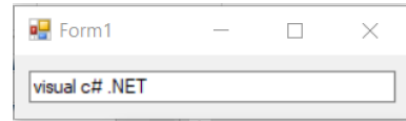
Bir metni belirtilen ayırıcı karaktere göre parçalara bölmek için kullanılır. Ayırıcı karakter olarak herhangi bir harf, özel karakter ( nokta,virgül gibi) kullanılabileceği gibi boşluk, Tab gibi ayırıcı karakterler de kullanılabilir.

**String\_değişken\_adı.Split('Karakter');**

Bir textBox'a girilen yazının kelime sayısını bulan program, split metodu kullanılarak yazılmak istenirse;

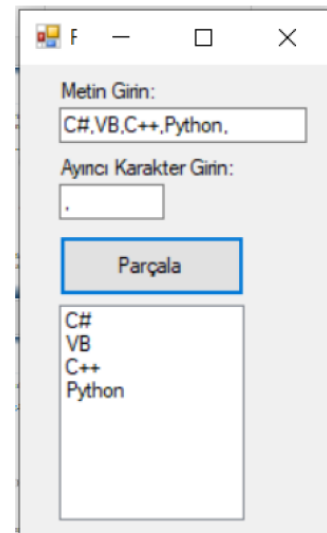
```
private void textBox1_KeyDown(object sender, KeyEventArgs e)
{
    string[] dizi;
    if (e.KeyCode==Keys.Enter)
    {
        dizi=textBox1.Text.Split(' ');
        MessageBox.Show("Cümlelerin Kelime Sayısı = " + dizi.Length.ToString());
    }
}
```

textBox1 içerisinde Enter tuşuna basıldığında, Split metodu kullanarak, cümlemizi boşluk karakterine göre parçalara ayırıp, Length metodu ile ayrılan parçaların sayısını bulunur.



## Split Metodu

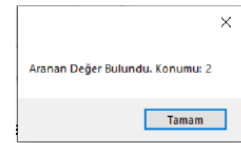
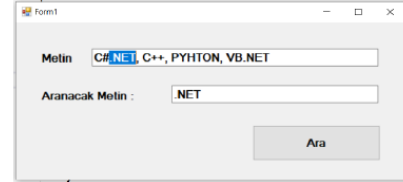
```
private void button1_Click(object sender, EventArgs e)
{
    byte i;
    string[] yazi;
    yazi = textBox1.Text.Split(Convert.ToChar(textBox2.Text));
    for(i=0;i<yazi.Length-1;i++)
    {
        listBox1.Items.Add(yazi[i].ToString());
    }
}
```



## IndexOf ve LastIndexOf Metodu

IndexOf ve LastIndexOf belirtilen karakter ya da stringin başka bir string içerisinde olup olmadığını arar ve eğer bulursa hangi konumda olduğunu bilgisini verir. Arama işlemini IndexOf baştan itibaren, LastIndexOf ise sondan itibaren yapar.

```
private void button1_Click(object sender, EventArgs e)
{
    int arama = txt_metin.Text.IndexOf(txt_aranan.Text);
    if (arama == 0)
        MessageBox.Show("Aranacak Değer Belirtilmedi!!!");
    else if (arama == -1)
        MessageBox.Show("Aranan Değer Bulunamadı!!!");
    else
    {
        txt_metin.SelectionStart = arama;
        txt_metin.SelectionLength = txt_aranan.Text.Length;
        txt_metin.Focus();
        MessageBox.Show("Aranan Değer Bulundu. Konumu: " + arama);
    }
}
```

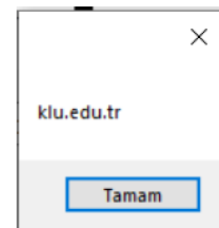
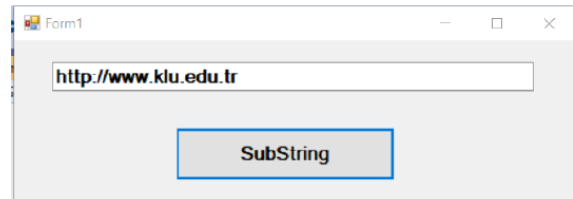


## SubString Metodu

SubString metodu bir metnin içerisinde belirtilen bir kısmını almak için kullanılır.

**String\_değişken\_adı.SubString(Başlangıç\_pozisyonu, uzunluk);**

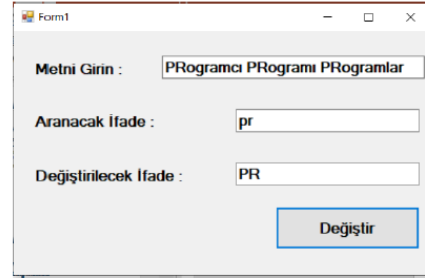
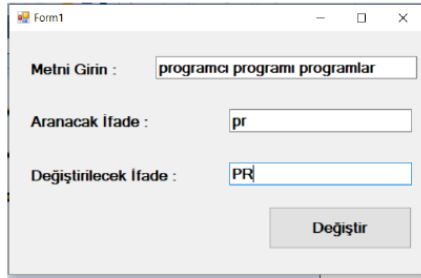
```
private void button1_Click(object sender, EventArgs e)
{
    int pozisyon, uzunluk;
    string cumle, cekilen;
    cumle = textBox1.Text;
    pozisyon = textBox1.Text.IndexOf(".") + 1;
    uzunluk = cumle.Length - pozisyon;
    cekilen = cumle.Substring(pozisyon, uzunluk);
    MessageBox.Show(cekilen);
}
```



## Replace Metodu

Bir metnin içerisinde, belirtilen bir ya da birden fazla karakteri, başka bir karakter ya da karakter ile değiştirmek için kullanılır.

**String\_değişken\_adı.Replace (eski\_değer, yeni\_değer);**



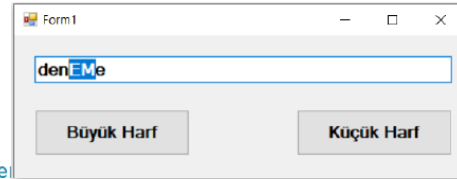
```
private void button1_Click(object sender, EventArgs e)
{
    textBox1.Text=textBox1.Text.Replace(textBox2.Text, textBox3.Text);
}
```

## ToUpper ve ToLower Metodu

ToUpper metodu bir string'i büyük harfe dönüştürürken, ToLower metodu ise küçük harfe dönüştürür.

**String\_değişken\_adı.ToUpper;**

**String\_değişken\_adı.ToLower;**



```
private void btn_buyuk_Click(object sender, EventArgs e)
{
    if (textBox1.SelectionLength > 0)
        textBox1.SelectedText = textBox1.SelectedText.ToUpper();
    else
        textBox1.Text = textBox1.Text.ToUpper();
}
```

1 reference

```
private void btn_kucuk_Click(object sender, EventArgs e)
{
    if (textBox1.SelectionLength > 0)
        textBox1.SelectedText = textBox1.SelectedText.ToLower();
    else
        textBox1.Text = textBox1.Text.ToLower();
}
```