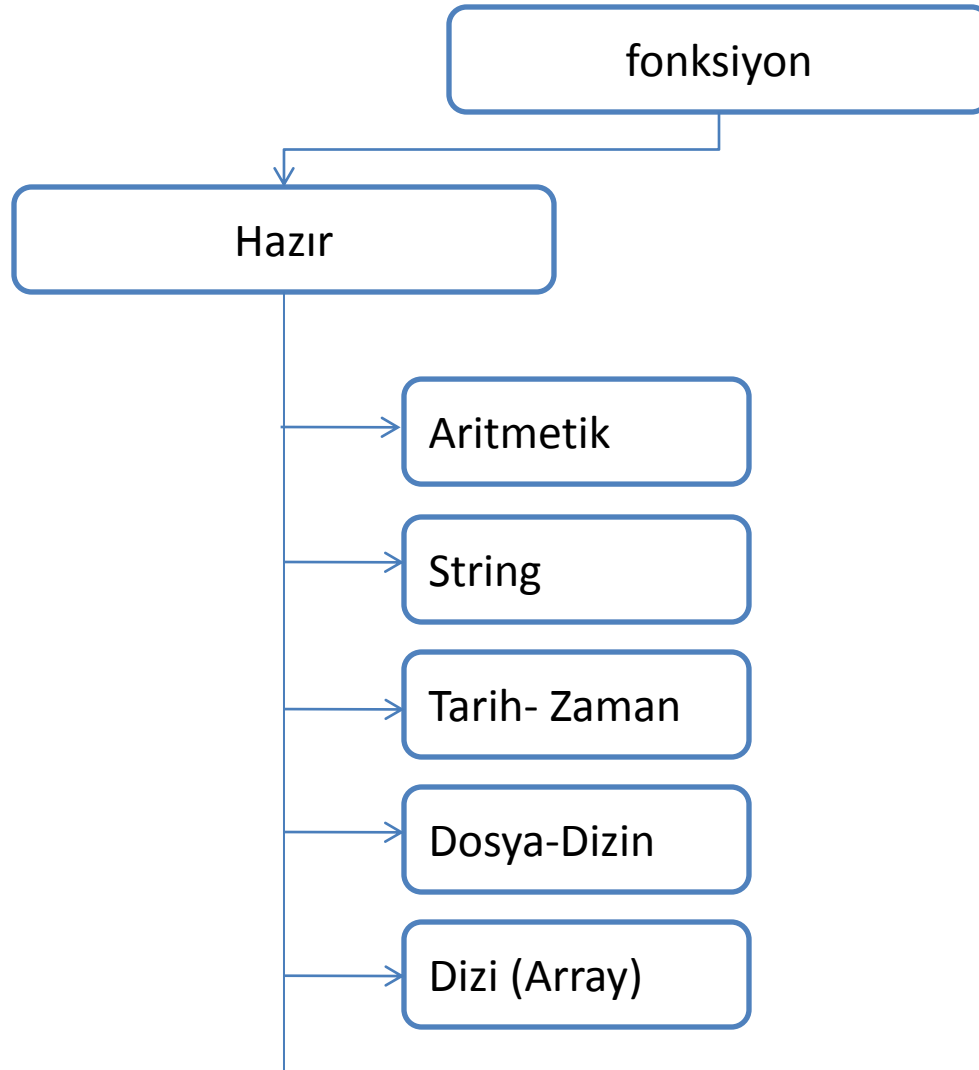


12- HAZIR FONKSİYONLAR



12- HAZIR FONKSİYONLAR

12.1 Aritmetik Fonksiyonlar

- ♣ `Math.Abs(reel sayı)` >> reel sayının pozitif değerini verir
- ♣ `Math.Ceiling(reel sayı)` >> reel sayıyı bir üst sayıya yuvarlar (6,2 -> 7)
- ♣ `Math.exp(reel sayı)` >> e^x ($e = \sim 2,71$)
- ♣ `Math.Floor (reel sayı)` >> reel sayıyı bir alt tam sayıya yuvarlar (9,8 -> 9)
- ♣ `Math.Log (sayı , taban)` >> $\log_{10} 50$
- ♣ `Math.Max(reel sayı1 , reel sayı2)` >> iki sayıdan büyük olanı verir
- ♣ `Math.Min(reel sayı1 , reel sayı2)` >> iki sayıdan küçük olanı verir
- ♣ `Math.PI` >> pi sayısını verir (22 / 7)
- ♣ `Math.E` >> e sayısını verir ($\sim 2,71$)
- ♣ `Math.Pow(taban , us)` >> x^y
- ♣ `Math.Round(reel sayı, hassasiyet)` >> ondalık kısmından kaç rakam gösterir.
- ♣ `Math.Sqrt (reel sayı)` >> sayının karekökünü alır
- ♣ `Math.Truncate (reel sayı)` >> Sayının tam kısmını alır (9,6 -> 9 , -5,7 -> -5)

12- HAZIR FONKSİYONLAR

double x;

- ♣ $x = \text{Math.Abs}(-8.4) ; // x = 8.2$
- ♣ $x = \text{Math.Ceiling}(-10.3) ; // x = -10$ büyüğe yuvarlar
- ♣ $x = \text{Math.Ceiling}(10.3) ; // x = 11$ büyüğe yuvarlar
- ♣ $x = \text{Math.Exp}(5) ; // x = e^5$
- ♣ $x = \text{Math.Floor}(12.8) ; // x = 12$ küçüğe yuvarlar
- ♣ $x = \text{Math.Floor}(-12.1) ; // x = -13$ küçüğe yuvarlar
- ♣ $x = \text{Math.Log}(125,5) ; // x = 3$ yani $\log_5 125$
- ♣ $x = \text{Math.Log10}(100) ; // x = 2$ yani $\log_{10} 100$
- ♣ $x = \text{Math.Max}(300,375) ; // x = 375$
- ♣ $x = \text{Math.Min}(300,375) ; // x = 300$
- ♣ $x = \text{Math.Round}(10.357812, 3) ; // x = 10.358$
- ♣ $x = \text{Math.Sign}(-3) ; // x = -1$
- ♣ $x = \text{Math.Sqrt}(16) ; // x = 4$
- ♣ $x = \text{Math.Pow}(3, 2) ; // x = 9$
- ♣ $x = \text{Math.PI} ; // x = 3.14$ pi sayısı
- ♣ $x = \text{Math.E} ; // x = e$ yani $x = 2.71$
- ♣ $x = \text{Math.Truncate}(8.9) ; // x = 8$
- ♣ $x = \text{Math.Truncate}(-8.9) ; // x = -8$

12- HAZIR FONKSİYONLAR

12.2 String Fonksiyonlar

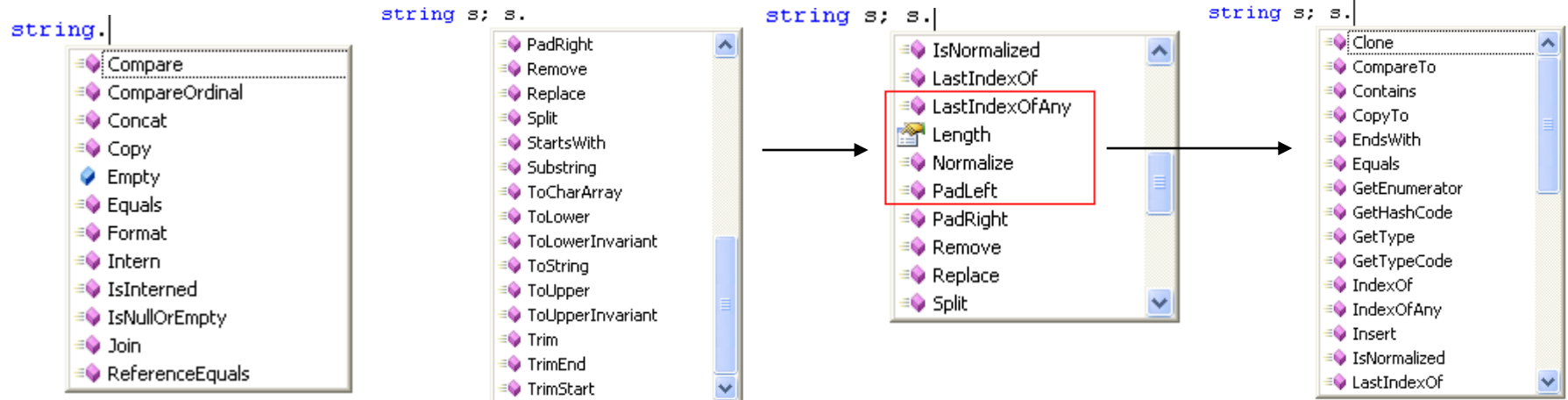
- ♣ `string.Compare(s1, s2) ;` //s2 içerisinde s1 'i arar varsa -1 eşitse 0 ve verir
- ♣ `string.Concat(s1,s2) ;` // s1 ve s2 stringlerini birleştirir
- ♣ `string.Equals(s1,s2) ;` // s1 ve s2 nin eşitliği test edilir eşitse *true*
- ♣ `string.Join (s1, dizi, x,son) ;` // s1 değeri dizinin x elemanına eklenir

`string s1,s2 ;`

- ♣ `k = s1.IndexOf(s2 , x , y) ;` // s1 içerisinde x den y 'ye kadar s2 aranır ve pozisyonu k' e aktarılır yoksa -1 aktarılır
- ♣ `s1.Insert (x,s2) ;` // s1 içerisine x' den itibaren s2 araya sokulur
- ♣ `k = s1.LastIndexOf (s2, x ,y) ;` // s1 içerisinde x den y 'ye kadar s2 aranır ve pozisyonu k' ya aktarılır yoksa -1 aktarılır. Burada arama işlemi sondan başa doğrudur.
- ♣ `k =s1.Length ;` // s1 in kaç karakter sayısını verir
- ♣ `s1.Remove (x ,y) ;` // s1 ' den x den sonra y adet karakter atılır
- ♣ `s1.Replace (m1,m2) ;` // s1 içerisinde m1 ile m2 stringi yer değiştirir
- ♣ `s2 = s1.Substring (x ,y) ;` // s1 den x den itibaren y kadar karakter kopyalanır
- ♣ `s1.ToUpper () ;` // s1 değişkenindeki harfler büyük harfe çevrilir
- ♣ `s1.ToLower () ;` // s1 değişkenindeki harfler küçük harfe çevrilir
- ♣ `s1.Trim () ;` // s1 değişkeninin başındaki ve sonundaki boşluklar atılır

12- HAZIR FONKSİYONLAR

12.2 String Fonksiyonlar



12- HAZIR FONKSİYONLAR

12.3 Tarih- Zaman Fonksiyonlar

DateTime x , y ; // x ve y değişkenleri tarih ve zaman tipindedir
x = Conert.ToDateTime(textBox1.Text) ; int k;

k = DateTime.Compare(x , y) ;	Eşitse k == 0 dır
k = DateTime.DaysInMonth(yil, ay);	yılın kaçınıcı ayı olduğunu verir
DateTime.IsLeapYer(yil) ;	Yil ' in artık yıl olup olmadığı (true/false)
DateTime.MaxValue;	Tarih ve zamanın alabileceği maximum değeri verir
DateTime.MinValue;	Tarih ve zamanın alabileceği minimum değeri verir
DateTime.Now ;	Sistemi aktif Tarih ve saatini verir
DateTime.Today ;	Sistemi aktif Tarih ve saatini verir (Zamanı 00:00:00 verir)

12- HAZIR FONKSİYONLAR

12.3 Tarih Zaman Fonksiyonlar

`DateTime x , y ; // x ve y değişkenleri tarih ve zaman tipindedir`
`x = Conert.ToDateTime(textBox1.Text) ; int k;`

<code>y = x.AddDays (gün) ;</code>	X tarihine gün ekler. Y de toplamı vardır
<code>y =x.AddMonth(ay) ;</code>	X tarihine ay ekler. Y de toplamı vardır
<code>y =x.AddYear(yıl) ;</code>	X tarihine yıl ekler. Y de toplamı vardır
<code>y = x.Day ;</code>	X tarihinden ayın gününü alır (0-31)
<code>y = x.DayOfWeek ;</code>	X tarihinden haftanın kaçınıcı günü olduğu (1 - 7)
<code>y = x.DayOfYear ;</code>	X tarihinden yılın kaçınıcı günü olduğu (1 – (365,366))
<code>y = x.Month ;</code>	X tarihinden yılın kaçınıcı ayı olduğu (1 – 12)
<code>y = x.Year ;</code>	X tarihinden yılı çekip alır
<code>y = x.ToLongDateString() ;</code>	X tarihini uzun tarih formatında (3 kasım 2004 Çarşamba) gibi
<code>y = x.ToShortDateString() ;</code>	X tarihini kısa tarih formatında (3.11.2004) gibi

12- HAZIR FONKSİYONLAR

12.3 Tarih Zaman Fonksiyonlar

```
DateTime x, y;  
int k;  
x = Convert.ToDateTime(Console.ReadLine() );  
y = Convert.ToDateTime(Console.ReadLine() );  
k = DateTime.Compare(x, y);  
if (k == 0) Console.Write ("\nEşit");  
else Console.Write ("\nFarklı");
```


12- HAZIR FONKSİYONLAR

12.3 Tarih Zaman Fonksiyonlar

```
DateTime x, y ;
int k ;
x = DateTime.Now;
string t1 = x.ToString();
string t2 = x.Date.ToString();
string t3 = x.Month.ToString();
string t4 = x.Year.ToString();
string t5 = x.ToLongDateString();
string t6 = x.Minute.ToString();
string t7 = x.DayOfWeek.ToString();
Console.Write (“{0}\n{1}\n{2}\n{3}\n{4}\n{5}\n{6}”, t1,t2,t3,t4,t5,t6,t7);
y = Convert.ToDateTime( Console.ReadLine() );
k = DateTime.Compare(x, y);
if (k == 0) Console.Write (“\nEşit”);
else Console.Write (“\nFarklı”);
```

12- HAZIR FONKSİYONLAR

12.4 Dizin Fonksiyonları

Using System.IO ; // ekleyiniz

- ♣ `Directory.Create (yol) ; Directory.Create(" C:\ \ VCS \ \ work ");`
- ♣ `Directory.Delete (yol) ; Directory.Delete(" C:\ \ VCS \ \ work ");`
- ♣ `Directory.Exists (yol) ;`
`if (Directory.Exists(" C:\ \ VCS \ \ work ") MessageBox.Show(" Dizin Mevcut");`
- ♣ `k = Directory.GetDirectories (yol) ; yol ile belirtilen dizin içindeki klasörlerin listesini k adlı string türündeki diziye aktarır`
`string[] dz = Directory.GetDirectories("c:\\Program Files");`
`foreach (string s in dz) label1.Text += s+"\n";`
- ♣ `Directory.GetFiles (yol) ; // yol ile belirtilen dizin içindeki dosyaların listesini k adlı string türündeki diziye aktarır`
`string[] ds = Directory.GetFiles("c:\\WINDOWS");`
`foreach (string s in ds) label2.Text += s + "\n";`
- ♣ `s = Directory.GetRootDirectory (yol) ; // yol ile belirtilen dizinin bağlı olduğu sürücüyü verir (string s ;)`

12- HAZIR FONKSİYONLAR

12.4 Dizin Fonksiyonları

- ♣ `k = Directory.GetLogicalDrives () ; // Bilgisayarınızdaki sürücülerin listesini k adlı string türündeki diziye aktarır`

```
string[ ] logicalDrive = Directory.GetLogicalDrives();  
foreach (string s in logicalDrive) label2.Text += s + "\n";
```

- ♣ `s = Directory.GetParent (yol) ; // yol ile belirttiğiniz dizinin bir üst dizinine geçmenizi sağlar`

```
string s;  
s = Directory.GetParent ( "c:\\ VCS \\ work" ) .ToString( ) ;  
label1.Text = s ; // Sonuç : c:\\ VCS  yazar
```

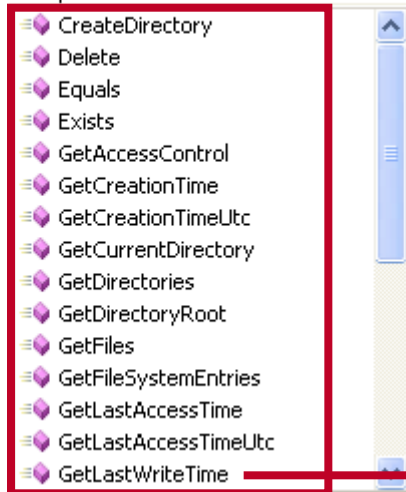
- ♣ `s = Directory.GetCurrentDirectory () ; // Çalışmakta olduğunuz akıt dizin
this.Text = Directory.GetCurrentDirectory();`

- ♣ `k = Directory.GetFileSystemEntries (dizin,dosya) ; // dizin içinde dosya ' yı arar ve bulursa
dosyanın yolu ile birlikte ismini k dizisine aktarır
string[] dosyadi = Directory.GetFileSystemEntries("d:\\","*.doc");
foreach (string s in dosyadi) label2.Text += s + "\n";`

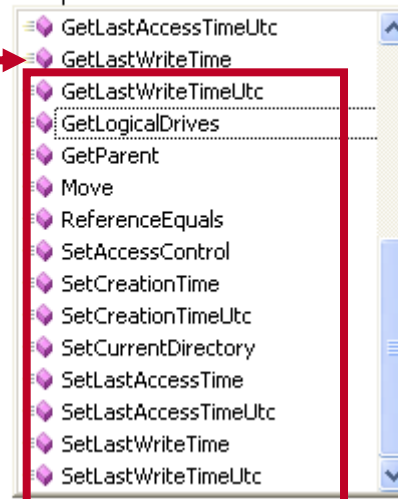
12- HAZIR FONKSİYONLAR

12.4 Dizin Fonksiyonları

```
string [] s;  
s = Directory.
```



```
string [] s;  
s = Directory.
```



12- HAZIR FONKSİYONLAR

12.4 Dosya- Dizin Fonksiyonları

```
using System.IO;

private void dizinlistele ( )
{
    string [ ] d=Directory.GetDirectories ("C:\\Program Files") ;
    foreach ( string kls in d)
        Console.Write ( "{0} \n" , kls ) ;
}
```

12- HAZIR FONKSİYONLAR

12.4 Dosya- Dizin Fonksiyonları

```
using System.IO;

private void dizinlistele ( )
{
    string [ ] d=Directory.GetFiles("C:\\ Program Files \\ FSI \\ F-Prot" ) ;
    foreach ( string kls in d)
        Console.Write ( "{0} \n" , kls ) ;
}
```

12- HAZIR FONKSİYONLAR

12.4 Dosya- Dizin Fonksiyonları

```
using System.IO;

private void dizinlistele ( )
{
    string [ ] d;
    d = Directory.GetFileSystemEntries("C:\\ WINDOWS \\ system32", "*.exe");
    foreach ( string kls in d)
        Console.Write ( "{0} \n" , kls ) ;
}
```

12- HAZIR FONKSİYONLAR

12.4 Dosya- Dizin Fonksiyonları

```
using System.IO;

private void dizinlistele ( )
{
    string [ ] s;
    s = Directory.GetLogicalDrives();
    foreach ( string src in s )
        Console.Write ( "{0} \n" , kls ) ;
}
```

A:\
B:\
C:\

12- HAZIR FONKSİYONLAR

12.4 Dosya Fonksiyonları

```
using System.IO;

private void dizinlistele ( )
{
    if (Directory.Exists("C:\\\\VCS") )
        Console.Write(" C:\\\\ VCS dizini mevcut " );
    else
    {
        Directory.CreateDirectory("C:\\\\VCS");
        Console.Write("C:\\\\VCS dizini OLUŞTURULDU ..." );
    }
    Console.Write ( "{0} \n" , kls ) ;
}
```

12- HAZIR FONKSİYONLAR

12.5 Dosyalama (Files) Fonksiyonları

```
string s;
```

```
FileInfo d = new FileInfo("C:\\ VCS \\ myo.dat ");
```

♣ `d.CopyTo(hedefdizin, true);` // d dosyasını hedef dizine kopyalar
`if(dosya.Exists) dosya.CopyTo("c:\\bolumilk.doc",true);`

♣ `d.Exists;` d dosyası mevcut mu? Örnl: `if(d.Exist) d.Delete ;`

♣ `d.Delete() ;` d dosyasını sil

♣ `d.DirectoryName ;` d dosyasının bulunduğu dizini verir (String türünden)
`this.Text = dosya.DirectoryName;`

♣ `d.Length) ;` d dosyasının byte cinsinden uzunluğunu verir
`this.Text = dosya.Length.ToString()+ " Byte";`

♣ `s = d.Name;` dosyanın sadece adını verir
`this.Text = dosya.Name;`

♣ `s = d.MoveTo(yol);` d dosyanı yol ile belirtilen yere taşır
`if (dosya.Exists) dosya.MoveTo("c:\\bolum1.doc");`

12- HAZIR FONKSİYONLAR

12.5 Dosyalama (Files) Fonksiyonları

```
using System.IO;
private void dosya_olustur ( )
{
    FileInfo d = new FileInfo("C:\\\\VCS\\\\bmyo.txt");
    // d.CopyTo("C:\\\\VCS\\\\bmyo.txt",true);
    label1.Text = d.Directory.ToString();

    label2.Text = d.DirectoryName;
    label3.Text = d.Length.ToString()+" Bytes" ;
    label4.Text = d.Name;
    // d.MoveTo("C:\\\\cs\\\\bmyo.txt");
    /*
    if (d.Exists) {
        d.Delete();
        MessageBox.Show("Dosya Silindi");
    } */
}
```

12- HAZIR FONKSİYONLAR

12.5 Dosyalama (Files) Fonksiyonları

```
using System.IO;
private void dosyadan_oku_yaz ( )
{
    string s;
    StreamReader okunan = new StreamReader("C:\\CS\\bmyo.txt");
    StreamWriter yazilan = new StreamWriter("C:\\CS\\egitim.txt");
    while ((s = okunan.ReadLine()) != null) yazilan.WriteLine(s);
    yazilan.Close();
    okunan.Close();
}
```

12- HAZIR FONKSİYONLAR

12.5 Dosyalama (Files) Fonksiyonları

```
using System.IO;
private void dosyadan_oku_yaz ( )
{
    string s;
    StreamReader dosya = File.OpenText("C:\\\\CS\\\\bmyo.txt");
    while ((s = dosya.ReadLine()) != null)
        Consoe.Write(s);
}
```

12- HAZIR FONKSİYONLAR

12.5 Dosyalama (Files) Fonksiyonları

```
using System.IO;
private void agdan_dosya_oku_yaz ( )
{
    string s;
    FileInfo f = new FileInfo(@" \\.\ L112_20 \\. CS \\. egitim.txt");
    StreamReader oku = new StreamReader("C:\\. CS \\. bmyo.txt");
    if ( f.Exists )
    {
        StreamWriter yaz = new StreamWriter(@" \\.\ L112_20 \\. CS \\. egitim.txt");
        while ((s = oku.ReadLine()) != null) yaz.WriteLine(s);
    }
    yaz.Close();
    oku.Close();
}
```

12- HAZIR FONKSİYONLAR

12.5 Dosyalama (Files) Fonksiyonları

```
using System.IO;
private void agdan_dosya_oku_yaz ( )
{
    string s;
    FileInfo f = new FileInfo((" \\ \\ L112_20 \\ CS \\ egitim.txt");
    StreamReader oku = new StreamReader (" \\ \\ L112_20 \\ CS \\ egitim.txt");
    if ( f.Exists )
    {
        StreamReader oku = new StreamReader(" \\ \\ L112_20 \\ CS \\ egitim.txt");
        while ((s = oku.ReadLine()) != null) Console.Write(s);
    }
    oku.Close();
}
```

Yerel ağdaki *L112_20* adlı bilgisayarda paylaşıma açılmış *CS* klasörü içerisinde bulunan *egitim.txt* dosyasını okuyup ekrana yazdırır.