

Açıklamalı kılavuz

Teyfik Aktaş tarafından hazırlanmış ve belgelendirilmiştir.

Künye:

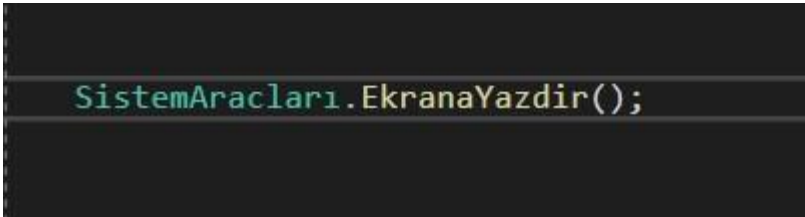
Fırat üniversitesi yazılım Mühendisliği 3.cü sınıf öğrencisi. Okul numarası:185541023

Tanım: Günümüzde iyiden iyiye herkesin öğrenmesi gereken bir dala dönüşen yazılımın herkese hitap etmesi için hazırlanmış tamamen Türkçe ve Kaba Kod odaklı tasarlanmış bir programlama kütüphanesidir. Halihazırda 4 temel işlem ve istenirse ileri matematiksel birkaç işlemi kapsamaktadır.



Sistem Araçları Sınıfı:

Sistem araçları sınıfı C# üzerinde bulunan System sınıfının basitleştirilmiş halidir.



Ekrana yazdır komutu:

C# konsol üzerinde yapılan işlemleri temel aldığı için bu kütüphanede de konsol üzerinden işlem yapmaktadır. İçerisine yazılan herhangi bir veri türünü ekrana yazacaktır.

```
SistemAracları.EkranaYazdir(8);
```

Sayısal Değerler bu şekilde ekrana yazılabilir.

```
8
```

Konsol ekranında bu şekil bir geri dönüş alınacaktır.

```
SistemAracları.EkranaYazdir("TÜRKÇE");
```

Metin Ağırlıklı kelimeler bu şekilde girilmelidir.

```
TÜRKÇE
```

Çıktısı bu şekilde olacaktır.

```
SistemAracları.HafizadaTut();
```

Çıkan sonuçları sonra kullanmak için hafızada tutmak isteyebilirsiniz. Bunun için Sistem araçları kütüphanesinde bulunan hafızada tut yöntemini kullanabilirsiniz. Aynı anda 4 sayıyı hafızada tutabilirsiniz ve istediğinizde tek tek veya tamamını silebilirsiniz. Siz Silmediğiniz sürece bu değişkenler program çalışmayı bitirene kadar hafızada kalacaktır.

```
SistemAracları.HafizadaTut(8);
```

Bu şekilde kullanımı yapılabilir .Bu değere ulaşmak için;

```
SistemAracları.Hafıza1;
```

Şeklinde ulaşılabilir. Yazdırmak içinse;

```
SistemAracları.EkranaYazdir(SistemAracları.Hafıza1);
```

Komutu kullanabilirsiniz.

```
SistemAracları.Hafıza1Temizle();
```

Bu şekilde hafıza 1 de tutulan değeri silebilirsiniz.
Tamamını silmek için ise;

```
SistemAracları.TümHafızalarıTemizle();
```

Diğer opsiyonlar için;



Yordamları Kullanılabilir.

Kullanıcıdan Değer almak için ;

```
Kullaniciİslemleri.DegerAl();
```

Kullanılabilir.

Bunu bir hafızaya atamak için;

```
SistemAracları.Hafıza1=Kullaniciİslemleri.DegerAl();
```

Bu yordam ile kullanıcıdan alınan değeri hafıza1 e atayacaktır. Başka bir yöntem ise

```
SistemAracları.HafizadaTut(Kullaniciİslemleri.DegerAl());
```

Kullanılabilir.

Matematik Sınıfı:

```
Matematikselİslemler
```

Temelde 4 temel işlevleri yerine getirir. Toplama çıkarma çarpma ve bölme harici ileri matematiksel işlevleri de yapmaktadır. İlerde bahsedilecektir. Bu yordamlarla yapılan tüm işlemler siz silmediğiniz sürece hafızada tutulacaktır. Silmek için gerekli yordam ilerde verecektir.

```
Matematikselİslemler.Toplama();
```

Toplama işlemlerini bu yordam ile yapabilirsiniz. Yaptığınız işlemler otomatik olarak hafızaya kayıt edilecektir. Üzerine yapılan işlemler bu sonuç üzerinden yapılacaktır.

```
Matematikselİslemler.Toplama(3,6,8);
```

Toplama işlemi yapacağınız değerleri yan yana virgül ile ayırarak yaptırabilirsiniz.

```
SistemAracları.EkranaYazdir(Matematikselİslemler.Toplama(3, 6, 8));
```

Bu şekilde hem toplayıp hem ekrana yazdırabilirsiniz. Yada herhangi bir hafızaya atayabilirsiniz.

Çıktı sonucu;

17

Olacaktır. Yan yana virgül ile ayırarak istediğiniz kadar değer toplayabilirsiniz.

```
Matematikselİslemler.Çıkarma(3, 6, 8);
```

Yine aynı şekilde çıkarma işlemi de yapabilirsiniz. Tüm özellikleri toplama ile aynıdır.

```
Matematikselİslemler.Çarpma(3, 6, 8);
```

Yine aynı şekilde çarpma işlemi de yapabilirsiniz. Tüm özellikleri toplama ile aynıdır.

```
Matematikselİslemler.Bölme(3, 6, 8);
```

Yine aynı şekilde bölme işlemi de yapabilirsiniz. Tüm özellikleri toplama ile aynıdır.

```
Matematikselİslemler.YüzdeAl();
```

İstenirse basit bir şekilde yüzde de alınabilir. Virgül ile ilk sıraya sayıyı ikini sayıya alınacak yüzdeyi yazabilirsiniz.

```
Matematikselİslemler.YüzdeAl(100,25);
```

Sonuç olarak çıktı bu olacaktır.

```
25
```

İstenirse bölme işleminde kalan da hızlı bir şekilde kullanılır.

```
Matematikselİslemler.KalanBul(100, 25);
```

İlk değere sayı bölünen ikinci sıraya bölen yazılırsa kalanı bulacaktır.

Sonuç :

```
0
```

Daha önce de bahsedildiği gibi matematiksel işlemler toplam değerinde program çalıştırdığı sürece değeri hafızada tutar. Silmek için;

```
Matematikselİslemler.ToplamTemizle();
```

Kullanılabilir.

Bu kütüphanede ileri bazı matematiksel işlemlerde kısa yoldan yapılabilir. Bu işlemler için;

```
Matematikselİslemler.İleriMatematikselİslemler
```

Kullanılabilir.

Karekök almak için;

```
Matematikselİslemler.İleriMatematikselİslemler.KareKokAl(25);
```

Kullanılabilir. Sonuç olarak;

5

Çıktısı verecektir.

10 tabanında logaritma alabilmek için;

```
Matematikselİslemler.İleriMatematikselİslemler.LogaritmaAl(25);
```

Kullanılabilir.

Üs almak için;

```
Matematikselİslemler.İleriMatematikselİslemler.ÜsAl(25,2);
```

Önce taban sayısını yazıp virgül konularak üs yazılabilir.

Çıktısı ;

625

Olacaktır.

Bir sayının mutlak değerini almak için ;

```
Matematikselİslemler.İleriMatematikselİslemler.MutlakDegerAl(-25);
```

Kullanılabilir Çıktısı ;

25

Olacaktır.

Son olarak bir sayının yuvarlamasını da alabilirsiniz.

Bunun için;

```
Matematikselİslemler.İleriMatematikselİslemler.YuvarlamaYap(3.4f);
```

Kullanılabilir. Son da belirtilen f harfi bu sayının virgüllü olduğunu belirtmek için kullanılmalıdır.

Çıktısı :

3

Değişken oluşturma sınıfı:

Şuan halihazırda üzerinde uğraşılan ve eklenmesi planlanan değişken oluşturma yordamları bulunmaktadır. Hala test aşamasında olduğu unutulmamalıdır.

```
DegiskenOlustur.TamSayiOlustur(8);  
DegiskenOlustur.KesirliSayiOlustur(7.3f);
```

Şeklinde kullanılabilir ve istenilen hafızaya kayıt edilebilir.