C Sharp Yardimci Metotlar

Tarih, Sayi, Metin, Sifreleme gibi konularda faydali olacak metotlar içeren bir Windows Form Uygulamasi

Table of Contents

Index   a

1 Symbol Reference

1.1 CSharpHelperMethods Namespace

This is namespace CSharpHelperMethods.

Namespaces

Name	Description
Library (⧉ see page 1)	This is namespace CSharpHelperMethods.Library.
Properties (⧉ see page 32)	This is namespace CSharpHelperMethods.Properties.
Test (⧉ see page 35)	This is namespace CSharpHelperMethods.Test.
UI (⧉ see page 45)	This is namespace CSharpHelperMethods.UI.

1.1.1 CSharpHelperMethods.Library Namespace

This is namespace CSharpHelperMethods.Library.

Classes

	Name	Description
⚙🅢	DataTableIslemleri (⧉ see page 2)	Raporlama için lazim olan DataTable vb. verilerin düzenlenmesini saglayan siniflardir.
⚙🅢	DosyaIslemleri (⧉ see page 4)	Dosya islemleri ile ilgili yardimici sinif
⚙🅢	KisiIslemleri (⧉ see page 8)	Kisi ile iliskili yardimci metotlarin tamamini içeren sinif
⚙🅢	MetinIslemleri (⧉ see page 10)	Metinsel veri kontrol metotlarini barindiran sinif
⚙🅢	Sabitler (⧉ see page 17)	Metin Sabitlerinin Bulundugu Sinif
⚙🅢	SayiIslemleri (⧉ see page 20)	Sayisal veri kontrol metotlarini barindiran sinif
⚙🅢	SifrelemeIslemleri (⧉ see page 23)	Sifre ile ilgili islemler sinifi
⚙🅢	SiralamaIslemleri (⧉ see page 25)	Alfa numerik siralama için hazirlanmis bir metod
⚙🅢	TarihIslemleri (⧉ see page 28)	Tarihsel veri kontrol metotlarini barindiran sinif

1.1.1.1 Classes

The following table lists classes in this documentation.

Classes

	Name	Description
⚙🅢	DataTableIslemleri (⧉ see page 2)	Raporlama için lazim olan DataTable vb. verilerin düzenlenmesini saglayan siniflardir.
⚙🅢	DosyaIslemleri (⧉ see page 4)	Dosya islemleri ile ilgili yardimici sinif
⚙🅢	KisiIslemleri (⧉ see page 8)	Kisi ile iliskili yardimci metotlarin tamamini içeren sinif
⚙🅢	MetinIslemleri (⧉ see page 10)	Metinsel veri kontrol metotlarini barindiran sinif

❖§	Sabitler (▨ see page 17)	Metin Sabitlerinin Bulundugu Sinif
❖§	SayiIslemleri (▨ see page 20)	Sayisal veri kontrol metotlarini barindiran sinif
❖§	SifrelemeIslemleri (▨ see page 23)	Sifre ile ilgili islemler sinifi
❖§	SiralamaIslemleri (▨ see page 25)	Alfa numerik siralama için hazirlanmis bir metod
❖§	TarihIslemleri (▨ see page 28)	Tarihsel veri kontrol metotlarini barindiran sinif

1.1.1.1.1 DataTableIslemleri Class

Raporlama için lazim olan DataTable vb. verilerin düzenlenmesini saglayan siniflardir.

Class Hierarchy

CSharpHelperMethods.Library.DataTableIslemleri

C#

```
public static class DataTableIslemleri;
```

File

DataTableIslemleri.cs (▨ see page 79)

DataTableIslemleri Methods

	Name	Description
➡❖§	BaseType (▨ see page 2)	Nesnenin içindeki her hangi bir degerin (DateTime, decimal vb.) Eger Nullable ise, Nullable olmamasini saglayan metottur (Çünkü Crystal Report, Telerik vb. Nullable alan kabul etmiyor)
➡❖	Nullable.GetUnderlyingType (▨ see page 2)	This is Nullable.GetUnderlyingType, a member of class DataTableIslemleri.
➡❖	oType.GetGenericTypeDefinition (▨ see page 3)	This is oType.GetGenericTypeDefinition, a member of class DataTableIslemleri.
➡§	ToDataTable (▨ see page 3)	Parametre olarak girilen herhangi bir sinif türünden listeyi Ideal bir sekilde Nullable alan içermeyen bir DataTable'a çevirmek için kullanilan metottur.
➡❖	typeof (▨ see page 4)	This is typeof, a member of class DataTableIslemleri.

1.1.1.1.1.1 DataTableIslemleri Methods

1.1.1.1.1.1.1 DataTableIslemleri.BaseType Method

Nesnenin içindeki her hangi bir degerin (DateTime, decimal vb.) Eger Nullable ise, Nullable olmamasini saglayan metottur (Çünkü Crystal Report, Telerik vb. Nullable alan kabul etmiyor)

C#

```
private static Type BaseType(Type oType);
```

Parameters

Parameters	Description
Type oType	Veri Tipi

Returns

Verinin Ideal Tipi

1.1.1.1.1.1.2 DataTableIslemleri.Nullable.GetUnderlyingType Method

C#

```
private Nullable.GetUnderlyingType(oType);
```

Description

This is Nullable.GetUnderlyingType, a member of class DataTableIslemleri.

1.1.1.1.1.1.3 DataTableIslemleri.oType.GetGenericTypeDefinition Method

C#

```
private >oType!=null&&oType.IsValueType&&oType.IsGenericType&&
oType.GetGenericTypeDefinition();
```

Description

This is oType.GetGenericTypeDefinition, a member of class DataTableIslemleri.

1.1.1.1.1.1.4 DataTableIslemleri.ToDataTable Method

Parametre olarak girilen herhangi bir sinif türünden listeyi Idael bir sekilde Nullable alan içermeyen bir DataTable'a çevirmek için kullanilan metottur.

C#

```
public static DataTable ToDataTable<T>(List<T> liste);
```

Parameters

Parameters	Description
List<T> liste	Liste Bilgisi

Type Parameters

Type Parameters	Description
T	Generic Nesne Tipi

Returns

DataTable cinsinde Nesne

Body Source

```
 1: public static DataTable ToDataTable<T>(List<T> liste)
 2: {
 3:     try
 4:     {
 5:         var sonuc = new DataTable(typeof(T).Name);
 6:
 7:         var ozellikler = typeof(T).GetFields(BindingFlags.Public |
BindingFlags.CreateInstance | BindingFlags.Default | BindingFlags.DeclaredOnly |
BindingFlags.ExactBinding | BindingFlags.FlattenHierarchy | BindingFlags.IgnoreReturn |
BindingFlags.Instance | BindingFlags.InvokeMethod | BindingFlags.NonPublic |
BindingFlags.OptionalParamBinding | BindingFlags.SuppressChangeType);
 8:
 9:         var ozellikler2 = typeof(T).GetProperties(BindingFlags.Public |
BindingFlags.CreateInstance | BindingFlags.Default | BindingFlags.DeclaredOnly |
BindingFlags.ExactBinding | BindingFlags.FlattenHierarchy | BindingFlags.IgnoreReturn |
BindingFlags.Instance | BindingFlags.InvokeMethod | BindingFlags.NonPublic |
BindingFlags.OptionalParamBinding | BindingFlags.SuppressChangeType);
10:
11:         foreach (var ozellik in ozellikler)
12:         {
13:             var varMi = false;
14:             foreach (var ozellik2 in ozellikler2)
15:             {
16:                 if (!ozellik.Name.Contains(ozellik2.Name)) continue;
17:                 varMi = true;
18:                 var ozellikAdi = ozellik.Name.Replace("<", "");
19:                 var ozellikDizi = ozellikAdi.Split('>');
20:                 var ozellikTemp = ozellikDizi[0].Trim();
```

```
21:                        //ozellikTemp = ozellikTemp.Substring(0, ozellikTemp.Length - 1);
22:                        if (ozellikTemp == ozellik2.Name)
23:                        {
24:                            if (sonuc.Columns.IndexOf(new DataColumn(ozellik2.Name)) >= 0)
continue;
25:                            sonuc.Columns.Add(ozellik2.Name, BaseType(ozellik2.PropertyType));
26:                            break;
27:                        }
28:
29:                        if (sonuc.Columns.IndexOf(new DataColumn(ozellikTemp)) >= 0) continue;
30:                        sonuc.Columns.Add(ozellikTemp, BaseType(ozellik2.PropertyType));
31:                        break;
32:                    }
33:
34:                    if (varMi) continue;
35:                    if (sonuc.Columns.IndexOf(new DataColumn(ozellik.Name)) < 0)
36:                    {
37:                        sonuc.Columns.Add(ozellik.Name, BaseType(ozellik.FieldType));
38:                    }
39:
40:                }
41:
42:                foreach (var nesne in liste)
43:                {
44:                    var degerler = new object[ozellikler.Length];
45:
46:                    for (var i = 0; i < ozellikler.Length; i++)
47:                    {
48:                        var varMi = false;
49:                        foreach (var item in ozellikler2)
50:                        {
51:                            if (!ozellikler[i].Name.Contains(item.Name)) continue;
52:                            varMi = true;
53:                            degerler[i] = item.GetValue(nesne, null);
54:                            break;
55:                        }
56:                        if (!varMi)
57:                        {
58:                            degerler[i] = ozellikler[i].GetValue(nesne);
59:                        }
60:
61:                    }
62:
63:                    sonuc.Rows.Add(degerler);
64:                }
65:
66:                return sonuc;
67:            }
68:        catch (Exception)
69:        {
70:            return null;
71:        }
72: }
```

1.1.1.1.1.1.5 DataTableIslemleri.typeof Method

C#

```
private typeof(Nullable<>);
```

Description

This is typeof, a member of class DataTableIslemleri.

1.1.1.1.2 DosyaIslemleri Class

Dosya islemleri ile ilgili yardimci sinif

Class Hierarchy

CSharpHelperMethods.Library.DosyaIslemleri

C#

```
public static class DosyaIslemleri;
```

File

DosyaIslemleri.cs (see page 81)

DosyaIslemleri Methods

	Name	Description
S	DosyaKilitliMi (see page 5)	Dosya sistemindeki bir dosyanin baska bir program tarafindan o anda kullanimda olup olmadiginin kontol edildigi bir metottur.
S	DosyaUzantisiUygunMu (see page 6)	Sisteme yüklenecek dosyanin belirlenen kriterlere uyup uymadigini kontrol eden metottur. Dosyanin uzantisi istedigimiz formatta degilse yanlis olarak geri döndürecektir.s
S	GetirDosyaUzantisiMimeTypeIle (see page 6)	Dosyanin türünden dosyanin uzantisinin ne oldugunu getiren metottur
S	GetirIcerikTipiDosyaUzantisiIle (see page 7)	Dosya uzantisindan dosyanin türünün ne oldugunu getiren metottur
S	GuvenliDizinOlustur (see page 7)	Bu metot ile gönderdiginiz Dizin yolunda verdiginiz isimde bir klasör yoksa Otomatik olarak olusturmayi sagliyor

1.1.1.1.2.1 DosyaIslemleri Methods

1.1.1.1.2.1.1 DosyaIslemleri.DosyaKilitliMi Method

Dosya sistemindeki bir dosyanin baska bir program tarafindan o anda kullanimda olup olmadiginin kontol edildigi bir metottur.

C#

```
public static bool DosyaKilitliMi(FileInfo dosya);
```

Parameters

Parameters	Description
FileInfo dosya	Dosya Bilgisi

Returns

Evet veya Hayir Bilgisi

Body Source

```
 1: public static bool DosyaKilitliMi(FileInfo dosya)
 2: {
 3:     FileStream stream = null;
 4:     try
 5:     {
 6:         stream = dosya.Open(FileMode.Open, FileAccess.ReadWrite, FileShare.None);
 7:     }
 8:     catch (IOException)
 9:     {
10:         // eger dosya baska bir uygulama tarafindan kullaniliyorsa
11:         // burada true dönüyor ki
12:         // kullanici bu metottan gelen bilgiye göre davransin
13:         return true;
14:     }
15:     finally
16:     {
```

```
17:          stream?.Close();
18:      }
19:
20:      //file is not locked
21:      return false;
22: }
```

1.1.1.1.2.1.2 DosyaIslemleri.DosyaUzantisiUygunMu Method

Sisteme yüklenecek dosyanin belirlenen kriterlere uyup uymadigini kontrol eden metottur. Dosyanin uzantisi istedigimiz formatta degilse yanlis olarak geri döndürecektir.s

C#

```
public static bool DosyaUzantisiUygunMu(string extension);
```

Parameters

Parameters	Description
string extension	Dosya Uzantisi

Returns

Eklenecek Dosyanin Uygun Olup Olmadigi Bilgisi

Body Source

```
1: public static bool DosyaUzantisiUygunMu(string extension)
2: {
3:      extension = extension.ToLower();
4:
5:      return extension == ".jpg" || extension == ".bmp" || extension == ".png" ||
extension == ".doc" || extension == ".docx" || extension == ".ppt" || extension == ".pptx"
|| extension == ".xls" || extension == ".xlsx" || extension == ".rar" || extension ==
".zip" || extension == ".7z" || extension == ".pdf" || extension == ".txt" || extension ==
".rtf";
6: }
```

1.1.1.1.2.1.3 DosyaIslemleri.GetirDosyaUzantisiMimeTypeIle Method

Dosyanin türünden dosyanin uzantisinin ne oldugunu getiren metottur

C#

```
public static string GetirDosyaUzantisiMimeTypeIle(string mimeType);
```

Parameters

Parameters	Description
string mimeType	Mime Type

Returns

Dosya Türü Bilgisi

Body Source

```
 1: public static string GetirDosyaUzantisiMimeTypeIle(string mimeType)
 2: {
 3:      string dosyaUzantisi;
 4:      mimeType = mimeType.ToLower();
 5:      switch (mimeType)
 6:      {
 7:          case "application/pdf": dosyaUzantisi = "pdf"; break;
 8:          case "image/jpeg": dosyaUzantisi = "jpeg"; break;
 9:          case "image/jpg": dosyaUzantisi = "jpg"; break;
10:          case "image/png": dosyaUzantisi = "png"; break;
11:          case "image/bmp": dosyaUzantisi = "bmp"; break;
12:          case "application/msword": dosyaUzantisi = "doc"; break;
13:          case "application / vnd.openxmlformats -
officedocument.presentationml.presentation": dosyaUzantisi = "pptx"; break;
```

```
14:          case "application/vnd.ms-excel": dosyaUzantisi = "xls"; break;
15:          case "text/plain": dosyaUzantisi = "txt"; break;
16:          case "application/vnd.openxmlformats-officedocument.spreadsheetml.sheet":
dosyaUzantisi = "xlsx"; break;
17:          case "application/vnd.openxmlformats-officedocument.wordprocessingml.document":
dosyaUzantisi = "docx"; break;
18:          case "text/xml": dosyaUzantisi = "xml"; break;
19:          case "application/vnd.ms-powerpoint": dosyaUzantisi = "ppt"; break;
20:          case "application/zip": dosyaUzantisi = "zip"; break;
21:          case "application/x-rar-compressed": dosyaUzantisi = "rar"; break;
22:          case "application/x-7z-compressed": dosyaUzantisi = "7z"; break;
23:          default: dosyaUzantisi = "error"; break;
24:      }
25:
26:      return dosyaUzantisi;
27: }
```

1.1.1.1.2.1.4 DosyaIslemleri.GetirIcerikTipiDosyaUzantisiIle Method

Dosya uzantisindan dosyanin türünün ne oldugunu getiren metottur

C#

```
public static string GetirIcerikTipiDosyaUzantisiIle(string extension);
```

Parameters

Parameters	Description
string extension	Dosya Uzantisi

Returns

Dosya Türü Bilgisi

Body Source

```
 1: public static string GetirIcerikTipiDosyaUzantisiIle(string extension)
 2: {
 3:      var contentType = String.Empty;
 4:      extension = extension.Replace(".", "").ToLower();
 5:      switch (extension)
 6:      {
 7:          case "pdf": contentType = "application/pdf"; break;
 8:          case "jpeg": contentType = "image/jpeg"; break;
 9:          case "jpg": contentType = "image/jpg"; break;
10:          case "png": contentType = "image/png"; break;
11:          case "bmp": contentType = "image/bmp"; break;
12:          case "doc": contentType = "application/msword"; break;
13:          case "pptx": contentType =
"application/vnd.openxmlformats-officedocument.presentationml.presentation"; break;
14:          case "xls": contentType = "application/vnd.ms-excel"; break;
15:          case "txt": contentType = "text/plain"; break;
16:          case "xlsx": contentType =
"application/vnd.openxmlformats-officedocument.spreadsheetml.sheet"; break;
17:          case "rtf": contentType = "application/rtf"; break;
18:          case "docx": contentType =
"application/vnd.openxmlformats-officedocument.wordprocessingml.document"; break;
19:          case "ppt": contentType = "application/vnd.ms-powerpoint"; break;
20:          case "zip": contentType = "application/zip"; break;
21:          case "rar": contentType = "application/x-rar-compressed"; break;
22:          case "7z": contentType = "application/x-7z-compressed"; break;
23:          default: contentType = "application/octet-stream"; break;
24:      }
25:
26:      return contentType;
27: }
```

1.1.1.1.2.1.5 DosyaIslemleri.GuvenliDizinOlustur Method

Bu metot ile gönderdiginiz Dizin yolunda verdiginiz isimde bir klasör yoksa Otomatik olarak olusturmayi sagliyor

C#

```
public static bool GuvenliDizinOlustur(string dizinYolu);
```

Parameters

Parameters	Description
string dizinYolu	Dizin Yolu Bilgisi

Returns

Evet veya Hayir Bilgisi

Body Source

```
1: public static bool GuvenliDizinOlustur(string dizinYolu)
2: {
3:     var isExists = Directory.Exists(dizinYolu);
4:
5:     if (!isExists)
6:         Directory.CreateDirectory(dizinYolu);
7:     return isExists;
8: }
```

1.1.1.1.3 KisiIslemleri Class

Kisi ile iliskili yardimci metotlarin tamamini içeren sinif

Class Hierarchy

CSharpHelperMethods.Library.KisiIslemleri

C#

```
public static class KisiIslemleri;
```

File

KisiIslemleri.cs (☐ see page 105)

KisiIslemleri Methods

	Name	Description
⬧⑤	DogrulaEPosta (☐ see page 8)	Ekranlardan girilen bir e-posta adresinin geçerli bir e-posta adresi olup olmadigini Kontrol eden metottur
⬧⑤	DogrulaIBAN (☐ see page 9)	Girilen IBAN'in standar IBAN algoritmasina uygun olup olmadigini dogrular
⬧⑤	DogrulaTcKimlikNo (☐ see page 10)	Girilen TC Kimlik Numarasinin, standart TC Kimlik No algoritmasina uygun olup olmadigini kontrol eden metottur.

1.1.1.1.3.1 KisiIslemleri Methods

1.1.1.1.3.1.1 KisiIslemleri.DogrulaEPosta Method

Ekranlardan girilen bir e-posta adresinin geçerli bir e-posta adresi olup olmadigini Kontrol eden metottur

C#

```
public static bool DogrulaEPosta(string ePosta);
```

Parameters

Parameters	Description
string ePosta	E-Posta Adresi

Returns

E-Posta Adresinin Geçerli Olup Olmadigi Bilgisi

Body Source

```
 1: public static bool DogrulaEPosta(string ePosta)
 2: {
 3:     const string pattern = @"^(([\w-]+\.)+[\w-]+|([a-zA-Z]{1}|[\w-]{2,}))@"
 4:                          + @"((([0-1]?[0-9]{1,2}|25[0-5]|2[0-4][0-9])\."
 5:                          + @"([0-1]?[0-9]{1,2}|25[0-5]|2[0-4][0-9])\."
 6:                          + @"([0-1]?[0-9]{1,2}|25[0-5]|2[0-4][0-9])\."
 7:                          + @"([0-1]?[0-9]{1,2}|25[0-5]|2[0-4][0-9])){1}|"
 8:                          + @"([a-zA-Z]+[\w-]+\.)+[a-zA-Z]{2,4})$";
 9:     return Regex.IsMatch(ePosta, pattern);
10: }
```

1.1.1.1.3.1.2 KisiIslemleri.DogrulaIBAN Method

Girilen IBAN'in standar IBAN algoritmasina uygun olup olmadigini dogrular

C#

```
public static bool DogrulaIBAN(string iban);
```

Parameters

Parameters	Description
string iban	Uluslar arasi banka hesap numarasi (IBAN)

Returns

IBAN uygunsa true, degilse false degeri döner

Description

```
DogrulaIBAN("TR560006200000012990022604")
```

Body Source

```
 1: public static bool DogrulaIBAN(string iban)
 2: {
 3:     iban = iban.ToUpper();
 4:
 5:     if (string.IsNullOrEmpty(iban))
 6:          return false;
 7:
 8:     if (!Regex.IsMatch(iban, "^[A-Z0-9]"))
 9:          return false;
10:
11:     iban = iban.Replace(" ", String.Empty);
12:     var bank = iban.Substring(4, iban.Length - 4) + iban.Substring(0, 4);
13:     const int asciiShift = 55;
14:     var sb = new StringBuilder();
15:     foreach (var c in bank)
16:     {
17:         sb.Append(char.IsLetter(c) ? c - asciiShift : int.Parse(c.ToString()));
18:     }
19:
20:     var checkSumString = sb.ToString();
21:     var checksum = int.Parse(checkSumString.Substring(0, 1));
22:     for (var i = 1; i < checkSumString.Length; i++)
23:     {
24:         var v = int.Parse(checkSumString.Substring(i, 1));
25:         checksum *= 10;
26:         checksum += v;
27:         checksum %= 97;
28:     }
29:     return checksum == 1;
30: }
```

1.1.1.1.3.1.3 KisiIslemleri.DogrulaTcKimlikNo Method

Girilen TC Kimlik Numarasinin, standart TC Kimlik No algoritmasina uygun olup olmadigini kontrol eden metottur.

C#

```
public static bool DogrulaTcKimlikNo(string tcKimlikNo);
```

Parameters

Parameters	Description
string tcKimlikNo	TC Kimlik No

Returns

TC Kimlik No'nun Uygun Olup Olmadigi Bilgisi

Body Source

```
 1: public static bool DogrulaTcKimlikNo(string tcKimlikNo)
 2: {
 3:     var tcNo = Convert.ToInt64(tcKimlikNo);
 4:     var atcno = tcNo / 100;
 5:     var btcno = tcNo / 100;
 6:     var c1 = atcno % 10; atcno /= 10;
 7:     var c2 = atcno % 10; atcno /= 10;
 8:     var c3 = atcno % 10; atcno /= 10;
 9:     var c4 = atcno % 10; atcno /= 10;
10:     var c5 = atcno % 10; atcno /= 10;
11:     var c6 = atcno % 10; atcno /= 10;
12:     var c7 = atcno % 10; atcno /= 10;
13:     var c8 = atcno % 10; atcno /= 10;
14:     var c9 = atcno % 10;
15:     var q1 = (10 - ((c1 + c3 + c5 + c7 + c9) * 3 + c2 + c4 + c6 + c8) % 10) % 10;
16:     var q2 = (10 - ((c2 + c4 + c6 + c8 + q1) * 3 + c1 + c3 + c5 + c7 + c9) % 10) % 10;
17:     var dogrula = btcno * 100 + q1 * 10 + q2 == tcNo;
18:     return dogrula;
19: }
```

1.1.1.1.4 MetinIslemleri Class

Metinsel veri kontrol metotlarini barindiran sinif

Class Hierarchy

CSharpHelperMethods.Library.MetinIslemleri

C#

```
public static class MetinIslemleri;
```

File

MetinIslemleri.cs (☐ see page 107)

MetinIslemleri Methods

	Name	Description
⇒◆S	BoslukKaldir (☐ see page 11)	Girilen metnin bosluklarinin kaldirilmasini saglayan metottur.
⇒◆ₐS	GereksizKarakterTemizle (☐ see page 11)	Telefon No, E-Posta Adresi vb. metinsel veriler içerisindeki Bosluk, parantez, kisa çizgi gibi verilerin silinmesi için hazirlanmis metottur
⇒◆S	HTMLDuzelt (☐ see page 12)	Girilen metnin basinda virgül olup olmadigini kontrol eden metottur.
⇒◆S	IdealMetinGetir (☐ see page 13)	Girilen metnin basinda virgül olup olmadigini kontrol eden metottur.
⇒◆S	MailAdresiGecerliMi (☐ see page 13)	Girilen mail adresinin geçerli olup olmadigini kontrol eden bir metottur

	MetinIlkKarakterBuyukDigerleriKucukYap (⊡ see page 13)	This is MetinIlkKarakterBuyukDigerleriKucukYap, a member of class MetinIslemleri.
⇒◆§	MetinIlkKarakterleriBuyukYap (⊡ see page 14)	Girilen bir metindeki tüm kelimelerin ilk harflerinin Büyük yapilmasi için hazirlanan metottur
⇒◆§	MetinSifrele (⊡ see page 14)	This is MetinSifrele, a member of class MetinIslemleri.
⇒◆§	MetinZararliKarakterDuzelt (⊡ see page 15)	This is MetinZararliKarakterDuzelt, a member of class MetinIslemleri.
⇒◆§	TelefonNoDuzenle (⊡ see page 15)	Girilen telefon numarasinin SMS gönderilecek hale getirilmesini saglayan metottur.
⇒◆₈	text.First (⊡ see page 15)	This is text.First, a member of class MetinIslemleri.
⇒◆₈	text.Substring (⊡ see page 15)	This is text.Substring, a member of class MetinIslemleri.
⇒◆₈	ToString (⊡ see page 16)	This is ToString, a member of class MetinIslemleri.
⇒◆₈	ToUpper (⊡ see page 16)	This is ToUpper, a member of class MetinIslemleri.
⇒◆§	TurkceKarakterleriDuzelt (⊡ see page 16)	Girilen metnin basinda virgül olup olmadigini kontrol eden metottur.
⇒◆§	UTF8Duzelt (⊡ see page 16)	Girilen metnin basinda virgül olup olmadigini kontrol eden metottur.

1.1.1.1.4.1 MetinIslemleri Methods

1.1.1.1.4.1.1 MetinIslemleri.BoslukKaldir Method

Girilen metnin bosluklarinin kaldirilmasini saglayan metottur.

C#

```
public static string BoslukKaldir(string metin);
```

Parameters

Parameters	Description
string metin	Telefon No Bilgisi

Returns

Girilen Metnin Istenen Hali

Body Source

```
1: public static string BoslukKaldir(string metin)
2: {
3:     if (string.IsNullOrEmpty(metin)) return string.Empty;
4:     metin = GereksizKarakterTemizle(metin);
5:     return metin;
6: }
```

1.1.1.1.4.1.2 MetinIslemleri.GereksizKarakterTemizle Method

Telefon No, E-Posta Adresi vb. metinsel veriler içerisindeki Bosluk, parantez, kisa çizgi gibi verilerin silinmesi için hazirlanmis metottur

C#

```
private static string GereksizKarakterTemizle(string metin);
```

Parameters

Parameters	Description
string metin	Girdi Metin Bilgisi

Returns

Çikti Metin Bilgisi

Body Source

```
1: private static string GereksizKarakterTemizle(string metin)
2: {
3:     return metin.Replace("-", "").Replace(" ", "").Replace("(", "").Replace(")", "");
4: }
```

1.1.1.1.4.1.3 MetinIslemleri.HTMLDuzelt Method

Girilen metnin basinda virgül olup olmadigini kontrol eden metottur.

C#

```
public static string HTMLDuzelt(string text);
```

Parameters

Parameters	Description
string text	Girdi metni

Returns

Girilen Metnin Olmasi Gerektigi Hali

Body Source

```
1: public static string HTMLDuzelt(string text)
2: {
3:     var sc = new List<string>
4:     {
5:         // get rid of unnecessary tag spans (comments and title)
6:         @"<!--(\w|\W)+?-->",
7:         @"<title>(\w|\W)+?</title>",
8:         // Get rid of classes and styles
9:         @"\s?class=\w+",
10:        @"\s+style='[^']+'",
11:        // Get rid of unnecessary tags
12:
@"<(meta|link|/?o:|/?style|/?div|/?st\d|/?head|/?html|body|/?body|/?span|!\[)[^>]*?>",
13:        // Get rid of empty paragraph tags
14:        @"(<[^>]+>)+ (</\w+>)+",
15:        @"<a[^>]*>([^<]+)</a>",
16:        @"<p[^>]*>([^<]+)</p>",
17:        // remove bizarre v: element attached to <img> tag
18:        @"\s+v:\w+=""[^""]+""",
19:        // remove extra lines
20:        @"(\n\r){2,}",
21:        " "
22:     };
23:     text = sc.Aggregate(text, (current, s) => Regex.Replace(current, s, string.Empty,
RegexOptions.IgnoreCase));
24:     text = Regex.Replace(text, "<.*?>", String.Empty);
25:     text = text.Replace("\r", " ").Replace("\n", " ").Replace("  ", " ").Trim();
26:     text = text.Replace("&rsquo;", "'");
27:     text = text.Replace("rsquo;", "'");
28:     text = text.Replace("&rdquo;", "\"");
29:     text = text.Replace("rdquo;", "\"");
30:
31:     text = text.Replace("&lsquo;", "'");
32:     text = text.Replace("lsquo;", "'");
33:     text = text.Replace("&ldquo;", "\"");
34:     text = text.Replace("ldquo;", "\"");
35:
36:     text = text.Replace("&lt;", "<");
37:     text = text.Replace("&gt;", ">");
38:     text = text.Replace("&amp;", "&");
39:     //text = text.Replace("&lt;", "");
40:     //text = text.Replace("p&gt;", "");
41:     //text = text.Replace("/p&gt;", "");
42:     //text = text.Replace("&gt;", "");
```

```
43:        //return text;
44:        return text;
45: }
```

1.1.1.1.4.1.4 MetinIslemleri.IdealMetinGetir Method

Girilen metnin basinda virgül olup olmadigini kontrol eden metottur.

C#

```
public static string IdealMetinGetir(string text);
```

Parameters

Parameters	Description
string text	Girdi metni

Returns

Girilen Metnin Olmasi Gerektigi Hali

Body Source

```
1: public static string IdealMetinGetir(string text)
2: {
3:     if (!string.IsNullOrEmpty(text) && text[0] == ',')
4:     {
5:         text = text.Substring(1, text.Length - 1);
6:     }
7:     return text;
8: }
```

1.1.1.1.4.1.5 MetinIslemleri.MailAdresiGecerliMi Method

Girilen mail adresinin geçerli olup olmadigini kontrol eden bir metottur

C#

```
public static bool MailAdresiGecerliMi(string email);
```

Parameters

Parameters	Description
string email	E-Posta Adresi

Returns

Geçerli veya Degil Bilgisi

Body Source

```
1: public static bool MailAdresiGecerliMi(string email)
2: {
3:     var gecerliMi = true;
4:     try
5:     {
6:         var emailAddress = new MailAddress(email);
7:     }
8:     catch
9:     {
10:         gecerliMi = false;
11:     }
12:     return gecerliMi;
13: }
```

1.1.1.1.4.1.6 MetinIslemleri.MetinIlkKarakterBuyukDigerleriKucukYap Method

C#

```
public static string MetinIlkKarakterBuyukDigerleriKucukYap(string text);
```

Description

This is MetinIlkKarakterBuyukDigerleriKucukYap, a member of class MetinIslemleri.

1.1.1.1.4.1.7 MetinIslemleri.MetinIlkKarakterleriBuyukYap Method

Girilen bir metindeki tüm kelimelerin ilk harflerinin Büyük yapilmasi için hazirlanan metottur

C#

```
public static string MetinIlkKarakterleriBuyukYap(this string input);
```

Parameters

Parameters	Description
this string input	Girilen metin

Returns

Metnin Kelimelerinin Ilk Harflerinin Büyültülmüs Hali

Body Source

```
 1: public static string MetinIlkKarakterleriBuyukYap(this string input)
 2: {
 3:     var metinDuzenleyici = new StringBuilder();
 4:     var wordList = input.Split(' ');
 5:     var cultureInfo = new CultureInfo("tr-TR");
 6:     foreach (var item in wordList)
 7:     {
 8:         if (item.Length > 1)
 9:         {
10:             metinDuzenleyici.Append(item.Substring(0, 1).ToUpper(cultureInfo));
11:             metinDuzenleyici.Append(item.Substring(1).ToLower(cultureInfo));
12:             metinDuzenleyici.Append(" ");
13:         }
14:         else if (item.Length == 1)
15:         {
16:             metinDuzenleyici.Append(item.ToUpper(cultureInfo));
17:             metinDuzenleyici.Append(" ");
18:         }
19:     }
20:     return metinDuzenleyici.ToString();
21: }
```

1.1.1.1.4.1.8 MetinIslemleri.MetinSifrele Method

C#

```
public static string MetinSifrele(string metin);
```

Description

This is MetinSifrele, a member of class MetinIslemleri.

Body Source

```
 1: public static string MetinSifrele(string metin)
 2: {
 3:     if (string.IsNullOrEmpty(metin)) return metin;
 4:     var metinDuzenleyici = new StringBuilder(metin);
 5:     switch (metin.Length)
 6:     {
 7:         case 10:
 8:             metinDuzenleyici.Remove(2, 6);
 9:             metinDuzenleyici.Insert(2, "******");
10:             metin = metinDuzenleyici.ToString();
11:             break;
12:         case 11:
13:             metinDuzenleyici.Remove(2, 7);
```

```
14:              metinDuzenleyici.Insert(2, "*******");
15:              metin = metinDuzenleyici.ToString();
16:              break;
17:      }
18:      return metin;
19: }
```

1.1.1.1.4.1.9 MetinIslemleri.MetinZararliKarakterDuzelt Method

C#

```
public static string MetinZararliKarakterDuzelt(string text);
```

Description

This is MetinZararliKarakterDuzelt, a member of class MetinIslemleri.

Body Source

```
1: public static string MetinZararliKarakterDuzelt(string text)
2: {
3:      text = text.Replace("&#252;", "ü");
4:      return text;
5: }
```

1.1.1.1.4.1.10 MetinIslemleri.TelefonNoDuzenle Method

Girilen telefon numarasinin SMS gönderilecek hale getirilmesini saglayan metottur.

C#

```
public static string TelefonNoDuzenle(string telefonNo);
```

Parameters

Parameters	Description
string telefonNo	Telefon No Bilgisi

Returns

Girilen Telefon Numarasinin SMS Gönderilecek Hali

Body Source

```
 1: public static string TelefonNoDuzenle(string telefonNo)
 2: {
 3:      if (string.IsNullOrEmpty(telefonNo)) return string.Empty;
 4:      telefonNo = GereksizKarakterTemizle(telefonNo);
 5:      if (telefonNo.Length == 11)
 6:      {
 7:          telefonNo = telefonNo.Substring(telefonNo.Length - 10);
 8:      }
 9:      return telefonNo;
10:
11: }
```

1.1.1.1.4.1.11 MetinIslemleri.text.First Method

C#

```
private > text.First();
```

Description

This is text.First, a member of class MetinIslemleri.

1.1.1.1.4.1.12 MetinIslemleri.text.Substring Method

C#

```
private text.Substring(1);
```

Description

This is text.Substring, a member of class MetinIslemleri.

1.1.1.1.4.1.13 MetinIslemleri.ToString Method

C#

```
private ToString();
```

Description

This is ToString, a member of class MetinIslemleri.

1.1.1.1.4.1.14 MetinIslemleri.ToUpper Method

C#

```
private ToUpper();
```

Description

This is ToUpper, a member of class MetinIslemleri.

1.1.1.1.4.1.15 MetinIslemleri.TurkceKarakterleriDuzelt Method

Girilen metnin basinda virgül olup olmadigini kontrol eden metottur.

C#

```
public static string TurkceKarakterleriDuzelt(string text);
```

Parameters

Parameters	Description
string text	Girdi metni

Returns

Girilen Metnin Olmasi Gerektigi Hali

Body Source

```
 1: public static string TurkceKarakterleriDuzelt(string text)
 2: {
 3:     text = text.Replace("ü", "u");
 4:     text = text.Replace("i", "i");
 5:     text = text.Replace("ö", "o");
 6:     text = text.Replace("ü", "u");
 7:     text = text.Replace("s", "s");
 8:     text = text.Replace("g", "g");
 9:     text = text.Replace("ç", "c");
10:     text = text.Replace("Ü", "U");
11:     text = text.Replace("I", "I");
12:     text = text.Replace("Ö", "O");
13:     text = text.Replace("Ü", "U");
14:     text = text.Replace("S", "S");
15:     text = text.Replace("G", "G");
16:     text = text.Replace("Ç", "C");
17:     return text;
18: }
```

1.1.1.1.4.1.16 MetinIslemleri.UTF8Duzelt Method

Girilen metnin basinda virgül olup olmadigini kontrol eden metottur.

C#

```
public static string UTF8Duzelt(string text);
```

Parameters

Parameters	Description
string text	Girdi metni

Returns

Girilen Metnin Olmasi Gerektigi Hali

Body Source

```
 1: public static string UTF8Duzelt(string text)
 2: {
 3:     text = text.Replace("&amp;ccedil;", "ç");
 4:     text = text.Replace("&amp;ouml;", "ö");
 5:     text = text.Replace("&amp;uuml;", "ü");
 6:     text = text.Replace("&ccedil;", "ç");
 7:     text = text.Replace("&ouml;", "ö");
 8:     text = text.Replace("&uuml;", "ü");
 9:     return text;
10: }
```

1.1.1.1.5 Sabitler Class

Metin Sabitlerinin Bulundugu Sinif

Class Hierarchy

CSharpHelperMethods.Library.Sabitler

C#

```
public static class Sabitler;
```

File

Sabitler.cs (see page 114)

Sabitler Fields

	Name	Description
♦	BaslangicTarihiBuyuk (see page 18)	This is BaslangicTarihiBuyuk, a member of class Sabitler.
♦	BaslangicTarihiGirilmedi (see page 18)	This is BaslangicTarihiGirilmedi, a member of class Sabitler.
♦	BaslangicVeyaBitisTarihiGirilmedi (see page 18)	public const string BitisTarihiGirilmedi = "Bitis Tarihi girilmedigi için bu islem gerçeklestirilemez.";
♦	EPostaAdresiGecerli (see page 18)	This is EPostaAdresiGecerli, a member of class Sabitler.
♦	EPostaAdresiGecerliDegil (see page 18)	This is EPostaAdresiGecerliDegil, a member of class Sabitler.
♦	MetinGirilmedi (see page 18)	This is MetinGirilmedi, a member of class Sabitler.
♦	MetinIbanNoDegil (see page 19)	This is MetinIbanNoDegil, a member of class Sabitler.
♦	MetinSayisalDegil (see page 19)	This is MetinSayisalDegil, a member of class Sabitler.
♦	MetinTcKimlikNoDegil (see page 19)	This is MetinTcKimlikNoDegil, a member of class Sabitler.
♦	MetinTcKimlikNoVeyaVergiNoDegil (see page 19)	This is MetinTcKimlikNoVeyaVergiNoDegil, a member of class Sabitler.
♦	SifrelenecekMetinGirilmedi (see page 19)	This is SifrelenecekMetinGirilmedi, a member of class Sabitler.
♦	SifreMetniGirilmedi (see page 19)	This is SifreMetniGirilmedi, a member of class Sabitler.

1.1.1.1.5.1 Sabitler Fields

1.1.1.1.5.1.1 Sabitler.BaslangicTarihiBuyuk Field

C#

```
public const string BaslangicTarihiBuyuk = "Baslangiç Tarihi Bitis Tarihinden büyük
olamaz.";
```

Description

This is BaslangicTarihiBuyuk, a member of class Sabitler.

1.1.1.1.5.1.2 Sabitler.BaslangicTarihiGirilmedi Field

C#

```
public const string BaslangicTarihiGirilmedi = "Baslangiç Tarihi girilmedigi için bu islem
gerçeklestirilemez.";
```

Description

This is BaslangicTarihiGirilmedi, a member of class Sabitler.

1.1.1.1.5.1.3 Sabitler.BaslangicVeyaBitisTarihiGirilmedi Field

C#

```
public const string BaslangicVeyaBitisTarihiGirilmedi = "Baslangiç Tarihi veya Bitis Tarihi
girilmedigi için bu islem gerçeklestirilemez.";
```

Description

public const string BitisTarihiGirilmedi = "Bitis Tarihi girilmedigi için bu islem gerçeklestirilemez.";

1.1.1.1.5.1.4 Sabitler.EPostaAdresiGecerli Field

C#

```
public const string EPostaAdresiGecerli = "Girilen e-posta adresi geçerli bir e-posta
adresidir";
```

Description

This is EPostaAdresiGecerli, a member of class Sabitler.

1.1.1.1.5.1.5 Sabitler.EPostaAdresiGecerliDegil Field

C#

```
public const string EPostaAdresiGecerliDegil = "Girilen e-posta adresi geçerli bir e-posta
adresi degildir";
```

Description

This is EPostaAdresiGecerliDegil, a member of class Sabitler.

1.1.1.1.5.1.6 Sabitler.MetinGirilmedi Field

C#

```
public const string MetinGirilmedi = "Metin girilmedigi için bu islem gerçeklestirilemez.";
```

Description

This is MetinGirilmedi, a member of class Sabitler.

1.1.1.1.5.1.7 **Sabitler.MetinIbanNoDegil Field**

C#

```
public const string MetinIbanNoDegil = "Girilen metin 26 karakter olmadigi için bu islem
gerçeklestirilemez.";
```

Description

This is MetinIbanNoDegil, a member of class Sabitler.

1.1.1.1.5.1.8 **Sabitler.MetinSayisalDegil Field**

C#

```
public const string MetinSayisalDegil = "Girilen metin sayisal bir deger olmadigi için bu
islem gerçeklestirilemez.";
```

Description

This is MetinSayisalDegil, a member of class Sabitler.

1.1.1.1.5.1.9 **Sabitler.MetinTcKimlikNoDegil Field**

C#

```
public const string MetinTcKimlikNoDegil = "Girilen metin 11 karakter olmadigi için bu
islem gerçeklestirilemez. Karakter Sayisi: ";
```

Description

This is MetinTcKimlikNoDegil, a member of class Sabitler.

1.1.1.1.5.1.10 **Sabitler.MetinTcKimlikNoVeyaVergiNoDegil Field**

C#

```
public const string MetinTcKimlikNoVeyaVergiNoDegil = "Girilen metin 10 veya 11 karakter
olmadigi için bu islem gerçeklestirilemez.";
```

Description

This is MetinTcKimlikNoVeyaVergiNoDegil, a member of class Sabitler.

1.1.1.1.5.1.11 **Sabitler.SifrelenecekMetinGirilmedi Field**

C#

```
public const string SifrelenecekMetinGirilmedi = "Sifrelenecek metin girilmedigi için bu
islem gerçeklestirilemez.";
```

Description

This is SifrelenecekMetinGirilmedi, a member of class Sabitler.

1.1.1.1.5.1.12 **Sabitler.SifreMetniGirilmedi Field**

C#

```
public const string SifreMetniGirilmedi = "Sifre metni girilmedigi için bu islem
gerçeklestirilemez.";
```

Description

This is SifreMetniGirilmedi, a member of class Sabitler.

1.1.1.1.6 SayiIslemleri Class

Sayisal veri kontrol metotlarini barindiran sinif

Class Hierarchy

CSharpHelperMethods.Library.SayiIslemleri

C#

```
public static class SayiIslemleri;
```

File

SayiIslemleri.cs (see page 115)

SayiIslemleri Methods

	Name	Description
⇒◆ S	AltiHaneliOnayKoduOlustur (see page 20)	Sadece rakamlardan olusan 6 haneli random sayi üreten metottur.
⇒◆ S	SayiHalineCevir (see page 20)	Ekrandan alinan string degerde virgül oldugundan ajax metodu patladigi için Burada string olarak tutar verisi aliniyor Ve decimala çevriliyor
⇒◆ S	SayisalMi (see page 21)	Girilen bir metnin sayi olup olmadigini kontrol eden metottur.
⇒◆ S	SayiyiYaziyaCevir (see page 21)	Elimizdeki tutar verilerinin Yazi formatinda gösterilmesini saglayacak bir metottur
⇒◆ ₈	text.All (see page 22)	This is text.All, a member of class SayiIslemleri.

1.1.1.1.6.1 SayiIslemleri Methods

1.1.1.1.6.1.1 SayiIslemleri.AltiHaneliOnayKoduOlustur Method

Sadece rakamlardan olusan 6 haneli random sayi üreten metottur.

C#

```
public static string AltiHaneliOnayKoduOlustur();
```

Returns

6 Haneli Onay Kodu

Body Source

```
1: public static string AltiHaneliOnayKoduOlustur()
2: {
3:     var rastgele = new Random();
4:     var sb = new StringBuilder();
5:     for (var i = 0; i < 6; i++)
6:     {
7:         var ascii = rastgele.Next(48, 57); //Rakamlar
8:         var karakter = Convert.ToChar(ascii);
9:         sb.Append(karakter);
10:    }
11:
12:    return sb.ToString();
13: }
```

1.1.1.1.6.1.2 SayiIslemleri.SayiHalineCevir Method

Ekrandan alinan string degerde virgül oldugundan ajax metodu patladigi için Burada string olarak tutar verisi aliniyor Ve decimala çevriliyor

C#

```
public static decimal SayiHalineCevir(string metin);
```

Parameters

Parameters	Description
string metin	Girilen String Deger

Returns

Decimal Deger

Body Source

```
 1: public static decimal SayiHalineCevir(string metin)
 2: {
 3:     decimal val;
 4:     if (metin.Contains("_") || metin.Contains(".") || metin.Contains("."))
 5:     {
 6:         if (!decimal.TryParse(metin.Replace(",", "").Replace(".", "").Replace("_", ""),
NumberStyles.Number,
 7:             CultureInfo.InvariantCulture, out val))
 8:             return 0;
 9:         return val / 100;
10:     }
11:
12:     var value = decimal.TryParse(metin.Replace(",", "").Replace(".", "").Replace("_",
""), NumberStyles.Number,
13:         CultureInfo.InvariantCulture, out val);
14:     return value ? val : 0;
15: }
```

1.1.1.1.6.1.3 SayiIslemleri.SayisalMi Method

Girilen bir metnin sayi olup olmadigini kontrol eden metottur.

C#

```
public static bool SayisalMi(string text);
```

Parameters

Parameters	Description
string text	Girdi metni

Returns

Girilen Metnin Sayisal Olup Olmadigi Bilgisi

1.1.1.1.6.1.4 SayiIslemleri.SayiyiYaziyaCevir Method

Elimizdeki tutar verilerinin Yazi formatinda gösterilmesini saglayacak bir metottur

C#

```
public static string SayiyiYaziyaCevir(decimal tutar);
```

Parameters

Parameters	Description
decimal tutar	Girilen Sayi Bilgisi

Returns

Sayinin Yaziya Dönüstürülmüs Hali

Body Source

```
 1: public static string SayiyiYaziyaCevir(decimal tutar)
```

```csharp
 2: {
 3:     var sTutar = tutar.ToString("F2").Replace('.', ','); // Replace('.',',') ondalik
ayracinin . olma durumu için
 4:     var lira = sTutar.Substring(0, sTutar.IndexOf(',')); //tutarin tam kismi
 5:     var kurus = sTutar.Substring(sTutar.IndexOf(',') + 1, 2);
 6:     var yazi = new StringBuilder();
 7:
 8:     var birler = new[] {"", "BIR", "IKI", "ÜÇ", "DÖRT", "BES", "ALTI", "YEDI", "SEKIZ",
"DOKUZ"};
 9:     var onlar = new[] {"", "ON", "YIRMI", "OTUZ", "KIRK", "ELLI", "ALTMIS", "YETMIS",
"SEKSEN", "DOKSAN"};
10:     var binler = new[] {"KENTRILYON", "KATRILYON", "TRILYON", "MILYAR", "MILYON",
"BIN", ""};
11:
12:     const int grupSayisi = 7;
13:     //sayidaki 3'lü grup sayisi. katrilyon içi 6. (1.234,00 daki grup sayisi 2'dir.)
14:     //KATRILYON'un basina ekleyeceginiz her deger için grup sayisini artiriniz.
15:
16:     lira = lira.PadLeft(grupSayisi * 3, '0'); //sayinin soluna '0' eklenerek sayi 'grup
sayisi x 3' basakmakli yapiliyor.
17:
18:     for (var i = 0; i < grupSayisi * 3; i += 3) //sayi 3'erli gruplar halinde ele
aliniyor.
19:     {
20:         var grupDegeri = new StringBuilder();
21:
22:         if (lira.Substring(i, 1) != "0")
23:             grupDegeri.Append(birler[Convert.ToInt32(lira.Substring(i, 1))] + "YÜZ");
//yüzler
24:         if (grupDegeri.ToString().Equals("BIRYÜZ"))
25:             grupDegeri.Replace("BIRYÜZ", "YÜZ");
26:         grupDegeri.Append(onlar[Convert.ToInt32(lira.Substring(i + 1, 1))]); //onlar
27:         grupDegeri.Append(birler[Convert.ToInt32(lira.Substring(i + 2,
1))]);//birler
28:         if (grupDegeri.Length > 0) //binler
29:             grupDegeri.Append(binler[i / 3]);
30:         if (grupDegeri.ToString().Equals("BIRBIN"))
31:             grupDegeri.Replace("BIRBIN", "BIN");
32:
33:         yazi.Append(grupDegeri);
34:     }
35:
36:     if (yazi.Length > 0)
37:         yazi.Append(" TL ");
38:
39:     var yaziUzunlugu = yazi.Length;
40:
41:     if (kurus.Substring(0, 1) != "0") //kurus onlar
42:         yazi.Append(onlar[Convert.ToInt32(kurus.Substring(0, 1))]);
43:
44:     if (kurus.Substring(1, 1) != "0") //kurus birler
45:         yazi.Append(birler[Convert.ToInt32(kurus.Substring(1, 1))]);
46:
47:     yazi.Append(yazi.Length > yaziUzunlugu ? " KR." : "SIFIR KR.");
48:
49:     return yazi.ToString();
50: }
```

1.1.1.1.6.1.5 SayiIslemleri.text.All Method

C#

```csharp
private > text.All(char. IsNumber);
```

Description

This is text.All, a member of class SayiIslemleri.

1.1.1.1.7 SifrelemeIslemleri Class

Sifre ile ilgili islemler sinifi

Class Hierarchy

CSharpHelperMethods.Library.SifrelemeIslemleri

C#

```
public static class SifrelemeIslemleri;
```

File

SifrelemeIslemleri.cs (see page 119)

SifrelemeIslemleri Methods

	Name	Description
S	HesaplaMD5 (see page 23)	Kullanici sifresinin MD5 yöntemiyle sifrelenmesini saglayan metottur.
S	HesaplaSHA256 (see page 23)	Kullanici sifresinin SHA yöntemiyle sifrelenmesini saglayan metottur.
S	HesaplaSHA512 (see page 24)	Kullanici sifresinin SHA yöntemiyle sifrelenmesini saglayan metottur.
S	RastgeleSifreUret (see page 24)	8 karakter uzunlugunda Rastgele sifre olusturur.
S	SifreUygunMu (see page 25)	sifreyi kriterlere göre kontrol edip uymayan kosullarini string listesine geri dönüs yapan methoddur. Uygun olmasi durumunda null döndürecektir.

1.1.1.1.7.1 SifrelemeIslemleri Methods

1.1.1.1.7.1.1 SifrelemeIslemleri.HesaplaMD5 Method

Kullanici sifresinin MD5 yöntemiyle sifrelenmesini saglayan metottur.

C#

```
public static string HesaplaMD5(string text);
```

Parameters

Parameters	Description
string text	Sifrelenecek Metin

Returns

Girdi Metninin Sifrelenmis Hali

Body Source

```
 1: public static string HesaplaMD5(string text)
 2: {
 3:     var md5 = MD5.Create();
 4:     var inputBytes = Encoding.ASCII.GetBytes(text);
 5:     var hash = md5.ComputeHash(inputBytes);
 6:
 7:     var sb = new StringBuilder();
 8:     foreach (var t in hash)
 9:     {
10:         sb.Append(t.ToString("X2"));
11:     }
12:     return sb.ToString();
13: }
```

1.1.1.1.7.1.2 SifrelemeIslemleri.HesaplaSHA256 Method

Kullanici sifresinin SHA yöntemiyle sifrelenmesini saglayan metottur.

C#

```
public static string HesaplaSHA256(string text);
```

Parameters

Parameters	Description
string text	Sifrelenecek Metin

Returns

Girdi Metninin Sifrelenmis Hali

Body Source

```
 1: public static string HesaplaSHA256(string text)
 2: {
 3:     var sha256 = SHA256.Create();
 4:     var bytes = Encoding.UTF8.GetBytes(text);
 5:     var hash = sha256.ComputeHash(bytes);
 6:     var result = new StringBuilder();
 7:     foreach (var t in hash)
 8:     {
 9:         result.Append(t.ToString("X2"));
10:     }
11:     return result.ToString();
12: }
```

1.1.1.1.7.1.3 SifrelemeIslemleri.HesaplaSHA512 Method

Kullanici sifresinin SHA yöntemiyle sifrelenmesini saglayan metottur.

C#

```
public static string HesaplaSHA512(string text);
```

Parameters

Parameters	Description
string text	Sifrelenecek Metin

Returns

Girdi Metninin Sifrelenmis Hali

Body Source

```
 1: public static string HesaplaSHA512(string text)
 2: {
 3:     var sha512 = new SHA512Managed();
 4:     var encryptedSha512 = sha512.ComputeHash(Encoding.UTF8.GetBytes(text));
 5:     var strHex = encryptedSha512.Aggregate("", (current, b) => $"{current}{b:x2}");
 6:     return strHex.ToUpper();
 7: }
```

1.1.1.1.7.1.4 SifrelemeIslemleri.RastgeleSifreUret Method

8 karakter uzunlugunda Rastgele sifre olusturur.

C#

```
public static string RastgeleSifreUret();
```

Returns

Sifrelenmis Metin Bilgisi

Body Source

```
 1: public static string RastgeleSifreUret()
 2: {
 3:     var sifreUzunluk = 8;
```

```
 4:        const string gecerliKarakterler = "abcdefghijklmnozABCDEFGHIJKLMNOZ1234567890";
 5:        var sifre = new StringBuilder(100);
 6:        var random = new Random();
 7:        while (0 < sifreUzunluk--)
 8:        {
 9:            sifre.Append(gecerliKarakterler[random.Next(gecerliKarakterler.Length)]);
10:        }
11:        return sifre.ToString();
12: }
```

1.1.1.1.7.1.5 SifrelemeIslemleri.SifreUygunMu Method

sifreyi kriterlere göre kontrol edip uymayan kosullarini string listesine geri dönüs yapan methoddur. Uygun olmasi durumunda null döndürecektir.

C#

```
public static string SifreUygunMu(string text);
```

Parameters

Parameters	Description
string text	Kullanici Sifresi

Returns

Sifrenin Uygun Olup Olmadigi Bilgisi (Açiklamasi ile beraber)

Body Source

```
 1: public static string SifreUygunMu(string text)
 2: {
 3:     //bu kisimda sifrenin olmasi gereken kosullari degerlendirilecek.
 4:     var mesajlar = new List<string>();
 5:
 6:     if (text.Length < 8)
 7:         mesajlar.Add("Parola en az 8 karakter olmalidir.");
 8:
 9:     if (!Regex.IsMatch(text, @"^(?=.*[a-z])", RegexOptions.ECMAScript))
10:         mesajlar.Add("Parola en az bir küçük harf içermelidir.");
11:
12:     if (!Regex.IsMatch(text, @"^(?=.*[A-Z])", RegexOptions.ECMAScript))
13:         mesajlar.Add("Parola en az bir büyük harf içermelidir.");
14:
15:     if (!Regex.IsMatch(text, @"^(?=.*[!,@,#,$,%,^,&,*,?,_,~,-,£,(,)])",
RegexOptions.ECMAScript))
16:         mesajlar.Add("Parola en az bir noktalama isareti içermelidir.");
17:
18:     if (!Regex.IsMatch(text, @"^(?=.*[0-9])", RegexOptions.ECMAScript))
19:         mesajlar.Add("Parola en az bir rakam içermelidir.");
20:
21:     if (mesajlar.Count < 1)
22:         return "Uygun";
23:     var sonuc = mesajlar.Aggregate("", (current, item) => current + item + ", ");
24:     sonuc = sonuc.Substring(0, sonuc.Length - 2);
25:     return sonuc;
26: }
```

1.1.1.1.8 SiralamaIslemleri Class

Alfa numerik siralama için hazirlanmis bir metod

Class Hierarchy

CSharpHelperMethods.Library.SiralamaIslemleri

C#

```
public static class SiralamaIslemleri;
```

File

SiralamaIslemleri.cs (⊡ see page 122)

SiralamaIslemleri Enumerations

	Name	Description
⬛	ChunkType (⊡ see page 26)	This is record CSharpHelperMethods.Library.SiralamaIslemleri.ChunkType.

SiralamaIslemleri Methods

	Name	Description
◆S	Compare (⊡ see page 26)	This is Compare, a member of class SiralamaIslemleri.
◆S	InChunk (⊡ see page 27)	This is InChunk, a member of class SiralamaIslemleri.

1.1.1.1.8.1 SiralamaIslemleri Enumerations

1.1.1.1.8.1.1 CSharpHelperMethods.Library.SiralamaIslemleri.ChunkType Enumeration

C#

```
private enum ChunkType {
  Alphanumeric,
  Numeric
}
```

File

SiralamaIslemleri.cs (⊡ see page 122)

Description

This is record CSharpHelperMethods.Library.SiralamaIslemleri.ChunkType.

1.1.1.1.8.2 SiralamaIslemleri Methods

1.1.1.1.8.2.1 SiralamaIslemleri.Compare Method

C#

```
public static int Compare(string x, string y);
```

Description

This is Compare, a member of class SiralamaIslemleri.

Body Source

```
 1: public static int Compare(string x, string y)
 2: {
 3:     if (IsNullOrEmpty(x) || IsNullOrEmpty(y)) return 0;
 4:     var thisMarker = 0;
 5:     var thatMarker = 0;
 6:
 7:     while (thisMarker < x.Length || thatMarker < y.Length)
 8:     {
 9:         if (thisMarker >= x.Length)
10:         {
11:             return -1;
12:         }
13:         if (thatMarker >= y.Length)
14:         {
15:             return 1;
16:         }
17:
```

```
18:             var thisCh = x[thisMarker];
19:             var thatCh = y[thatMarker];
20:
21:             var thisChunk = new StringBuilder();
22:             var thatChunk = new StringBuilder();
23:
24:             while (thisMarker < x.Length && (thisChunk.Length == 0 || InChunk(thisCh,
thisChunk[0])))
25:             {
26:                 thisChunk.Append(thisCh);
27:                 thisMarker++;
28:
29:                 if (thisMarker < x.Length)
30:                 {
31:                     thisCh = x[thisMarker];
32:                 }
33:             }
34:
35:             while (thatMarker < y.Length && (thatChunk.Length == 0 || InChunk(thatCh,
thatChunk[0])))
36:             {
37:                 thatChunk.Append(thatCh);
38:                 thatMarker++;
39:
40:                 if (thatMarker < y.Length)
41:                 {
42:                     thatCh = y[thatMarker];
43:                 }
44:             }
45:
46:             var result = 0;
47:             // If both chunks contain numeric characters, sort them numerically
48:             if (char.IsDigit(thisChunk[0]) && char.IsDigit(thatChunk[0]))
49:             {
50:                 var thisNumericChunk = Convert.ToInt64(thisChunk.ToString());
51:                 var thatNumericChunk = Convert.ToInt64(thatChunk.ToString());
52:
53:                 if (thisNumericChunk < thatNumericChunk)
54:                 {
55:                     result = -1;
56:                 }
57:
58:                 if (thisNumericChunk > thatNumericChunk)
59:                 {
60:                     result = 1;
61:                 }
62:             }
63:             else
64:             {
65:                 var ci = new CultureInfo("tr-TR");
66:                 result = String.Compare(thisChunk.ToString(), thatChunk.ToString(), true,
67:                     ci);
68:                 //result = thisChunk.ToString().CompareTo(thatChunk.ToString());
69:             }
70:
71:             if (result == 0) continue;
72:             return result;
73:         }
74:     return 0;
75: }
```

1.1.1.1.8.2.2 SiralamaIslemleri.InChunk Method

C#

```
private static bool InChunk(char ch, char otherCh);
```

Description

This is InChunk, a member of class SiralamaIslemleri.

Body Source

```
 1: private static bool InChunk(char ch, char otherCh)
 2: {
 3:     var type = ChunkType.Alphanumeric;
 4:
 5:     if (char.IsDigit(otherCh))
 6:     {
 7:         type = ChunkType.Numeric;
 8:     }
 9:
10:     return (type != ChunkType.Alphanumeric || !char.IsDigit(ch)) && (type !=
ChunkType.Numeric || char.IsDigit(ch));
11: }
```

1.1.1.1.9 TarihIslemleri Class

Tarihsel veri kontrol metotlarini barindiran sinif

Class Hierarchy

CSharpHelperMethods.Library.TarihIslemleri

C#

```
public static class TarihIslemleri;
```

File

TarihIslemleri.cs (▣ see page 124)

TarihIslemleri Methods

	Name	Description
⇒◈**S**	AyFarki (▣ see page 28)	Girilen iki tarih arasinda kaç ay oldugunu bulan metottur
⇒◈**S**	KidemTazminatiTarihAraligiHesapla (▣ see page 29)	Kidem Tazminati Hesaplamasi Yapilirken ayda kaç gün olursa olsun tamaminda çalismissa 30 gün üzerinden hesaplanmasi gerektigi için Isten ayrilis ve is baslangiç tarihleri arasinda kaç gün geçtigini belirtmek için Hesaplama yapilmasini saglayan metottur.
⇒◈	Math.Abs (▣ see page 30)	This is Math.Abs, a member of class TarihIslemleri.
⇒◈**S**	TarihAraligiHesapla (▣ see page 30)	Iki tarih arasinda kaç gün oldugunu hesaplamak için kullanilan metottur
⇒◈**S**	YasHesapla (▣ see page 30)	Yas hesaplamak için kullanilan metottur
⇒◈**S**	YasHesaplaMetinsel (▣ see page 31)	Yas hesaplamak için kullanilan metottur
⇒◈**S**	ZamaniDegistir (▣ see page 32)	Filtre ekranlarinda baslangiç ve bitis tarihi üzerinden sorgulama yapildiginda Eger ayni gün için sorgulama yapiliyorsa Bitis tarihini o günün 23.59.59.999'una ayarlamak için Hazirlanan metottur

1.1.1.1.9.1 TarihIslemleri Methods

1.1.1.1.9.1.1 TarihIslemleri.AyFarki Method

Girilen iki tarih arasinda kaç ay oldugunu bulan metottur

C#

```
public static int AyFarki(DateTime lValue, DateTime rValue);
```

Parameters

Parameters	Description
DateTime lValue	Ilk tarih bilgisi
DateTime rValue	Son tarih bilgisi

Returns

Ay Farki Bilgisi

1.1.1.1.9.1.2 **TarihIslemleri.KidemTazminatiTarihAraligiHesapla Method**

Kidem Tazminati Hesaplamasi Yapilirken ayda kaç gün olursa olsun tamaminda çalismissa 30 gün üzerinden hesaplanmasi gerektigi için Isten ayrilis ve is baslangiç tarihleri arasinda kaç gün geçtigini belirtmek için Hesaplama yapilmasini saglayan metottur.

C#

```
public static int KidemTazminatiTarihAraligiHesapla(DateTime? baslangicTarihi, DateTime?
bitisTarihi);
```

Parameters

Parameters	Description
DateTime? baslangicTarihi	Baslangiç Tarihi Bilgisi
DateTime? bitisTarihi	Bitis Tarihi Bilgisi

Returns

Gün Sayisi Bilgisi

Body Source

```
 1: public static int KidemTazminatiTarihAraligiHesapla(DateTime? baslangicTarihi,
DateTime? bitisTarihi)
 2: {
 3:     var artacakMi = true;
 4:     var result = 0;
 5:     var baslangicMi = false;
 6:     var bitisMi = false;
 7:     var aylikGunSayisi = 0;
 8:     for (var date = (DateTime)baslangicTarihi; date <= (DateTime)bitisTarihi; date =
date.AddDays(1))
 9:     {
10:         if (!artacakMi)
11:             result++;
12:         else
13:             artacakMi = false;
14:
15:         if (date.Day == 1)
16:             baslangicMi = true;
17:         else if (date.Day == DateTime.DaysInMonth(date.Year, date.Month))
18:             bitisMi = true;
19:
20:         if (baslangicMi)
21:             aylikGunSayisi++;
22:         if (baslangicMi && bitisMi && aylikGunSayisi != 30)
23:         {
24:             result = result - aylikGunSayisi + 30;
25:             baslangicMi = false;
26:             bitisMi = false;
27:             aylikGunSayisi = 0;
28:         }
29:         else if (bitisMi)
30:         {
31:             baslangicMi = false;
32:             bitisMi = false;
33:             aylikGunSayisi = 0;
34:         }
35:
36:     }
37:     if (baslangicTarihi.Value.Day > bitisTarihi.Value.Day)
38:         result--;
39:     return result;
```

```
40: }
```

1.1.1.1.9.1.3 TarihIslemleri.Math.Abs Method

C#

```
private > Math.Abs(lValue.Month - rValue.Month + 12 * (lValue.Year - rValue.Year));
```

Description

This is Math.Abs, a member of class TarihIslemleri.

1.1.1.1.9.1.4 TarihIslemleri.TarihAraligiHesapla Method

Iki tarih arasinda kaç gün oldugunu hesaplamak için kullanilan metottur

C#

```
public static string TarihAraligiHesapla(DateTime baslangicTarihi, DateTime bitisTarihi);
```

Parameters

Parameters	Description
DateTime baslangicTarihi	Baslangiç Tarihi Bilgisi
DateTime bitisTarihi	Bitis Tarihi Bilgisi

Returns

Tarih Araligi Bilgisi

Body Source

```
 1: public static string TarihAraligiHesapla(DateTime baslangicTarihi, DateTime bitisTarihi)
 2: {
 3:     var years = new DateTime(bitisTarihi.Subtract(baslangicTarihi).Ticks).Year - 1;
 4:     var pastYearDate = baslangicTarihi.AddYears(years);
 5:     var months = 0;
 6:     for (var i = 1; i <= 12; i++)
 7:     {
 8:         if (pastYearDate.AddMonths(i) == bitisTarihi)
 9:         {
10:             months = i;
11:             break;
12:         }
13:
14:         if (pastYearDate.AddMonths(i) < bitisTarihi) continue;
15:         months = i - 1;
16:         break;
17:     }
18:     var days = bitisTarihi.Subtract(pastYearDate.AddMonths(months)).Days;
19:     var result = new StringBuilder();
20:     result.Append(years.ToString().PadLeft(2, '0'));
21:     result.Append(" Yil, ");
22:     result.Append(months.ToString().PadLeft(2, '0'));
23:     result.Append(" Ay, ");
24:     result.Append(days.ToString().PadLeft(2, '0'));
25:     result.Append(" Gün");
26:     return result.ToString();
27: }
```

1.1.1.1.9.1.5 TarihIslemleri.YasHesapla Method

Yas hesaplamak için kullanilan metottur

C#

```
public static int YasHesapla(DateTime dogumTarihi);
```

Parameters

Parameters	Description
DateTime dogumTarihi	Dogum Tarihi Bilgisi

Returns

Yas Bilgisi

Body Source

```csharp
 1: public static int YasHesapla(DateTime dogumTarihi)
 2: {
 3:     var now = DateTime.Now;
 4:     var years = new DateTime(DateTime.Now.Subtract(dogumTarihi).Ticks).Year - 1;
 5:     var pastYearDate = dogumTarihi.AddYears(years);
 6:     for (var i = 1; i <= 12; i++)
 7:     {
 8:         if (pastYearDate.AddMonths(i) == now)
 9:           break;
10:         if (pastYearDate.AddMonths(i) < now) continue;
11:         break;
12:     }
13:     return years;
14: }
```

1.1.1.1.9.1.6 TarihIslemleri.YasHesaplaMetinsel Method

Yas hesaplamak için kullanilan metottur

C#

```csharp
public static string YasHesaplaMetinsel(DateTime dogumTarihi);
```

Parameters

Parameters	Description
DateTime dogumTarihi	Dogum Tarihi Bilgisi

Returns

Yas Bilgisi (Metinsel olarak)

Body Source

```csharp
 1: public static string YasHesaplaMetinsel(DateTime dogumTarihi)
 2: {
 3:     var now = DateTime.Now;
 4:     var years = new DateTime(DateTime.Now.Subtract(dogumTarihi).Ticks).Year - 1;
 5:     var pastYearDate = dogumTarihi.AddYears(years);
 6:     var months = 0;
 7:     for (var i = 1; i <= 12; i++)
 8:     {
 9:         if (pastYearDate.AddMonths(i) == now)
10:         {
11:             months = i;
12:             break;
13:         }
14:
15:         if (pastYearDate.AddMonths(i) < now) continue;
16:         months = i - 1;
17:         break;
18:     }
19:     var days = now.Subtract(pastYearDate.AddMonths(months)).Days;
20:     // var hours = now.Subtract(pastYearDate).Hours;
21:     // var minutes = now.Subtract(pastYearDate).Minutes;
22:     // var seconds = now.Subtract(pastYearDate).Seconds;
23:     return $"Yas: {years} Yil {months} Ay {days} Gün";
24:     //return String.Format("Yas: {0} Yil {1} Ay {2} Gün {3} Saat {4} Saniye",
25:     //years, months, Days, Hours, Seconds);
```

```
26: }
```

1.1.1.1.9.1.7 TarihIslemleri.ZamaniDegistir Method

Filtre ekranlarinda baslangiç ve bitis tarihi üzerinden sorgulama yapildiginda Eger ayni gün için sorgulama yapiliyorsa Bitis tarihini o günün 23.59.59.999'una ayarlamak için Hazirlanan metottur

C#

```csharp
public static DateTime ZamaniDegistir(this DateTime dateTime, int hours, int minutes, int
seconds, int milliseconds);
```

Parameters

Parameters	Description
this DateTime dateTime	Tarih Bilgisi
int hours	Saat Bilgisi
int minutes	Dakika Bilgisi
int seconds	Saniye Bilgisi
int milliseconds	Mili Saniye Bilgisi

Returns

Yeni Tarih Bilgisi

Body Source

```csharp
 1: public static DateTime ZamaniDegistir(this DateTime dateTime, int hours, int minutes,
int seconds, int milliseconds)
 2: {
 3:     return new DateTime(
 4:         dateTime.Year,
 5:         dateTime.Month,
 6:         dateTime.Day,
 7:         hours,
 8:         minutes,
 9:         seconds,
10:         milliseconds,
11:         dateTime.Kind);
12: }
```

1.1.2 CSharpHelperMethods.Properties Namespace

This is namespace CSharpHelperMethods.Properties.

Classes

	Name	Description
❖	Resources (☐ see page 33)	A strongly-typed resource class, for looking up localized strings, etc.
❖	Settings (☐ see page 34)	This is class CSharpHelperMethods.Properties.Settings.

1.1.2.1 Classes

The following table lists classes in this documentation.

Classes

	Name	Description
❖	Resources (☐ see page 33)	A strongly-typed resource class, for looking up localized strings, etc.
❖	Settings (☐ see page 34)	This is class CSharpHelperMethods.Properties.Settings.

1.1.2.1.1 Resources Class

A strongly-typed resource class, for looking up localized strings, etc.

Class Hierarchy

CSharpHelperMethods.Properties.Resources

C#

```
[global::System.CodeDom.Compiler.GeneratedCodeAttribute("System.Resources.Tools.StronglyType
dResourceBuilder",
 "4.0.0.0")]
[global::System.Diagnostics.DebuggerNonUserCodeAttribute()]
[global::System.Runtime.CompilerServices.CompilerGeneratedAttribute()]
internal class Resources;
```

File

Resources.Designer.cs (⊡ see page 113)

Description

This class was auto-generated by the StronglyTypedResourceBuilder class via a tool like ResGen or Visual Studio. To add or remove a member, edit your .ResX file then rerun ResGen with the /str option, or rebuild your VS project.

Methods

	Name	Description
≡◆♦	Resources (⊡ see page 33)	This is Resources, a member of class Resources.

Resources Fields

	Name	Description
♦ₐ§	resourceCulture (⊡ see page 34)	This is resourceCulture, a member of class Resources.
♦ₐ§	resourceMan (⊡ see page 34)	This is resourceMan, a member of class Resources.

Resources Properties

	Name	Description
☰♦§	Culture (⊡ see page 34)	Overrides the current thread's CurrentUICulture property for all resource lookups using this strongly typed resource class.
☰♦§	ResourceManager (⊡ see page 34)	Returns the cached ResourceManager instance used by this class.

1.1.2.1.1.1 Resources.Resources Constructor

C#

```
[global::System.Diagnostics.CodeAnalysis.SuppressMessageAttribute("Microsoft.Performance",
 "CA1811:AvoidUncalledPrivateCode")]
internal Resources();
```

Description

This is Resources, a member of class Resources.

Body Source

```
1:
[global::System.Diagnostics.CodeAnalysis.SuppressMessageAttribute("Microsoft.Performance",
 "CA1811:AvoidUncalledPrivateCode")]
2: internal Resources()
3: {
4: }
```

1.1.2.1.1.2 Resources Fields

1.1.2.1.1.2.1 Resources.resourceCulture Field

C#

```
private static global::System.Globalization.CultureInfo resourceCulture;
```

Description

This is resourceCulture, a member of class Resources.

1.1.2.1.1.2.2 Resources.resourceMan Field

C#

```
private static global::System.Resources.ResourceManager resourceMan;
```

Description

This is resourceMan, a member of class Resources.

1.1.2.1.1.3 Resources Properties

1.1.2.1.1.3.1 Resources.Culture Property

Overrides the current thread's CurrentUICulture property for all resource lookups using this strongly typed resource class.

C#

```
[global::System.ComponentModel.EditorBrowsableAttribute(global::System.ComponentModel.Editor
BrowsableState.Advanced)]
internal static global::System.Globalization.CultureInfo Culture;
```

1.1.2.1.1.3.2 Resources.ResourceManager Property

Returns the cached ResourceManager instance used by this class.

C#

```
[global::System.ComponentModel.EditorBrowsableAttribute(global::System.ComponentModel.Editor
BrowsableState.Advanced)]
internal static global::System.Resources.ResourceManager ResourceManager;
```

1.1.2.1.2 Settings Class

Class Hierarchy

```
global::System.Configuration.ApplicationSettingsBase  →  CSharpHelperMethods.Properties.Settings
```
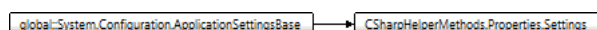
C#

```
[global::System.Runtime.CompilerServices.CompilerGeneratedAttribute()]
[global::System.CodeDom.Compiler.GeneratedCodeAttribute("Microsoft.VisualStudio.Editors.Sett
ingsDesigner.SettingsSingleFileGenerator",
"11.0.0.0")]
internal sealed class Settings : global::System.Configuration.ApplicationSettingsBase;
```

File

Settings.Designer.cs (▣ see page 118)

Description

This is class CSharpHelperMethods.Properties.Settings.

Settings Fields

	Name	Description
♦ⓈⓈ	defaultInstance (▣ see page 35)	This is defaultInstance, a member of class Settings.

Settings Properties

	Name	Description
🖻🅂	Default (🖻 see page 35)	This is Default, a member of class Settings.

1.1.2.1.2.1 Settings Fields

1.1.2.1.2.1.1 Settings.defaultInstance Field

C#

```
private static Settings defaultInstance =
((Settings)(global::System.Configuration.ApplicationSettingsBase.Synchronized(new
Settings()))));
```

Description

This is defaultInstance, a member of class Settings.

1.1.2.1.2.2 Settings Properties

1.1.2.1.2.2.1 Settings.Default Property

C#

```
public static Settings Default;
```

Description

This is Default, a member of class Settings.

1.1.3 CSharpHelperMethods.Test Namespace

This is namespace CSharpHelperMethods.Test.

Classes

	Name	Description
🔩	MetinIslemleriTest (🖻 see page 36)	Library (🖻 see page 1) kisminda Metin Islemleri bölümünde Yer alan metotlarin örnek test durumlari ve beklenen sonuçlari ile Test edildikleri siniftir
🔩	SayiIslemleriTest (🖻 see page 40)	Library (🖻 see page 1) kisminda Sayi Islemleri bölümünde Yer alan metotlarin örnek test durumlari ve beklenen sonuçlari ile Test edildikleri siniftir
🔩	SifrelemeIslemleriTest (🖻 see page 41)	Library (🖻 see page 1) kisminda Sifreleme Islemleri bölümünde Yer alan metotlarin örnek test durumlari ve beklenen sonuçlari ile Test edildikleri siniftir
🔩	TarihIslemleriTest (🖻 see page 43)	Library (🖻 see page 1) kisminda Tarih Islemleri bölümünde Yer alan metotlarin örnek test durumlari ve beklenen sonuçlari ile Test edildikleri siniftir

1.1.3.1 Classes

The following table lists classes in this documentation.

Classes

	Name	Description
🔩	MetinIslemleriTest (⊠ see page 36)	Library (⊠ see page 1) kisminda Metin Islemleri bölümünde Yer alan metotlarin örnek test durumlari ve beklenen sonuçlari ile Test (⊠ see page 35) edildikleri siniftir
🔩	SayiIslemleriTest (⊠ see page 40)	Library (⊠ see page 1) kisminda Sayi Islemleri bölümünde Yer alan metotlarin örnek test durumlari ve beklenen sonuçlari ile Test (⊠ see page 35) edildikleri siniftir
🔩	SifrelemeIslemleriTest (⊠ see page 41)	Library (⊠ see page 1) kisminda Sifreleme Islemleri bölümünde Yer alan metotlarin örnek test durumlari ve beklenen sonuçlari ile Test (⊠ see page 35) edildikleri siniftir
🔩	TarihIslemleriTest (⊠ see page 43)	Library (⊠ see page 1) kisminda Tarih Islemleri bölümünde Yer alan metotlarin örnek test durumlari ve beklenen sonuçlari ile Test (⊠ see page 35) edildikleri siniftir

1.1.3.1.1 MetinIslemleriTest Class

Library (⊠ see page 1) kisminda Metin Islemleri bölümünde Yer alan metotlarin örnek test durumlari ve beklenen sonuçlari ile Test (⊠ see page 35) edildikleri siniftir

Class Hierarchy

CSharpHelperMethods.Test.MetinIslemleriTest

C#

```
[TestFixture]
public class MetinIslemleriTest;
```

File

MetinIslemleriTest.cs (⊠ see page 111)

MetinIslemleriTest Methods

	Name	Description
⇒◆	BoslukKaldirTest (⊠ see page 37)	This is BoslukKaldirTest, a member of class MetinIslemleriTest.
⇒◆	IdealMetinGetirTest (⊠ see page 37)	This is IdealMetinGetirTest, a member of class MetinIslemleriTest.
⇒◆	MailAdresiGecerliMiTest (⊠ see page 37)	This is MailAdresiGecerliMiTest, a member of class MetinIslemleriTest.
⇒◆	MetinIlkKarakterBuyukDigerleriKucukYapTest (⊠ see page 38)	This is MetinIlkKarakterBuyukDigerleriKucukYapTest, a member of class MetinIslemleriTest.
⇒◆	MetinIlkKarakterleriBuyukYapTest (⊠ see page 38)	This is MetinIlkKarakterleriBuyukYapTest, a member of class MetinIslemleriTest.
⇒◆	MetinSifreleTest (⊠ see page 38)	This is MetinSifreleTest, a member of class MetinIslemleriTest.
⇒◆	MetinZararliKarakterDuzeltTest (⊠ see page 39)	This is MetinZararliKarakterDuzeltTest, a member of class MetinIslemleriTest.
⇒◆	TelefonNoDuzenleTest (⊠ see page 39)	This is TelefonNoDuzenleTest, a member of class MetinIslemleriTest.
⇒◆	TurkceKarakterleriDuzeltTest (⊠ see page 39)	This is TurkceKarakterleriDuzeltTest, a member of class MetinIslemleriTest.

1.1.3.1.1.1 MetinIslemleriTest Methods

1.1.3.1.1.1.1 **MetinIslemleriTest.BoslukKaldirTest Method**

C#

```
[Test]
[TestCase("2013123817  ", ExpectedResult = "2013123817")]
[TestCase("ses ss", ExpectedResult = "sesss")]
public string BoslukKaldirTest(string metin);
```

Description

This is BoslukKaldirTest, a member of class MetinIslemleriTest.

Body Source

```
1: [Test]
2: [TestCase("2013123817  ", ExpectedResult = "2013123817")]
3: [TestCase("ses ss", ExpectedResult = "sesss")]
4: public string BoslukKaldirTest(string metin)
5: {
6:     return MetinIslemleri.BoslukKaldir(metin);
7: }
```

1.1.3.1.1.1.2 **MetinIslemleriTest.IdealMetinGetirTest Method**

C#

```
[Test]
[TestCase("serdar", ExpectedResult = "serdar")]
[TestCase(",test metni", ExpectedResult = "test metni")]
public string IdealMetinGetirTest(string metin);
```

Description

This is IdealMetinGetirTest, a member of class MetinIslemleriTest.

Body Source

```
1: [Test]
2: [TestCase("serdar", ExpectedResult = "serdar")]
3: [TestCase(",test metni", ExpectedResult = "test metni")]
4: public string IdealMetinGetirTest(string metin)
5: {
6:     return MetinIslemleri.IdealMetinGetir(metin);
7: }
```

1.1.3.1.1.1.3 **MetinIslemleriTest.MailAdresiGecerliMiTest Method**

C#

```
[Test]
[TestCase("serdargul@outlook.com", ExpectedResult = true)]
[TestCase("serdar@aol.com", ExpectedResult = true)]
[TestCase("a.b", ExpectedResult = false)]
[TestCase("a_c", ExpectedResult = false)]
public bool MailAdresiGecerliMiTest(string metin);
```

Description

This is MailAdresiGecerliMiTest, a member of class MetinIslemleriTest.

Body Source

```
1: [Test]
2: [TestCase("serdargul@outlook.com", ExpectedResult = true)]
3: [TestCase("serdar@aol.com", ExpectedResult = true)]
4: [TestCase("a.b", ExpectedResult = false)]
5: [TestCase("a_c", ExpectedResult = false)]
6: public bool MailAdresiGecerliMiTest(string metin)
7: {
8:     return MetinIslemleri.MailAdresiGecerliMi(metin);
9: }
```

1.1.3.1.1.1.4 MetinIslemleriTest.MetinIlkKarakterBuyukDigerleriKucukYapTest Method

C#

```
[Test]
[TestCase("serdar", ExpectedResult = "Serdar")]
[TestCase("bu BIR test metnidir", ExpectedResult = "Bu BIR test metnidir")]
public string MetinIlkKarakterBuyukDigerleriKucukYapTest(string metin);
```

Description

This is MetinIlkKarakterBuyukDigerleriKucukYapTest, a member of class MetinIslemleriTest.

Body Source

```
1: [Test]
2: [TestCase("serdar", ExpectedResult = "Serdar")]
3: [TestCase("bu BIR test metnidir", ExpectedResult = "Bu BIR test metnidir")]
4: public string MetinIlkKarakterBuyukDigerleriKucukYapTest(string metin)
5: {
6:     return MetinIslemleri.MetinIlkKarakterBuyukDigerleriKucukYap(metin);
7: }
```

1.1.3.1.1.1.5 MetinIslemleriTest.MetinIlkKarakterleriBuyukYapTest Method

C#

```
[Test]
[TestCase("serdar", ExpectedResult = "Serdar ")]
[TestCase("bu BIR test metnidir", ExpectedResult = "Bu Bir Test Metnidir ")]
public string MetinIlkKarakterleriBuyukYapTest(string metin);
```

Description

This is MetinIlkKarakterleriBuyukYapTest, a member of class MetinIslemleriTest.

Body Source

```
1: [Test]
2: [TestCase("serdar", ExpectedResult = "Serdar ")]
3: [TestCase("bu BIR test metnidir", ExpectedResult = "Bu Bir Test Metnidir ")]
4: public string MetinIlkKarakterleriBuyukYapTest(string metin)
5: {
6:     return MetinIslemleri.MetinIlkKarakterleriBuyukYap(metin);
7: }
```

1.1.3.1.1.1.6 MetinIslemleriTest.MetinSifreleTest Method

C#

```
[Test]
[TestCase("2013123817", ExpectedResult = "20******17")]
[TestCase("11111111111", ExpectedResult = "11*******11")]
[TestCase("serdar", ExpectedResult = "serdar")]
public string MetinSifreleTest(string metin);
```

Description

This is MetinSifreleTest, a member of class MetinIslemleriTest.

Body Source

```
1: [Test]
2: [TestCase("2013123817", ExpectedResult = "20******17")]
3: [TestCase("11111111111", ExpectedResult = "11*******11")]
4: [TestCase("serdar", ExpectedResult = "serdar")]
5: public string MetinSifreleTest(string metin)
6: {
7:     return MetinIslemleri.MetinSifrele(metin);
8: }
```

1.1.3.1.1.1.7 MetinIslemleriTest.MetinZararliKarakterDuzeltTest Method

C#

```
[Test]
[TestCase("2013123817", ExpectedResult = "2013123817")]
[TestCase("sesss&#252;", ExpectedResult = "sesssü")]
public string MetinZararliKarakterDuzeltTest(string metin);
```

Description

This is MetinZararliKarakterDuzeltTest, a member of class MetinIslemleriTest.

Body Source

```
1: [Test]
2: [TestCase("2013123817", ExpectedResult = "2013123817")]
3: [TestCase("sesss&#252;", ExpectedResult = "sesssü")]
4: public string MetinZararliKarakterDuzeltTest(string metin)
5: {
6:     return MetinIslemleri.MetinZararliKarakterDuzelt(metin);
7: }
```

1.1.3.1.1.1.8 MetinIslemleriTest.TelefonNoDuzenleTest Method

C#

```
[Test]
[TestCase("0(505)333-44-55", ExpectedResult = "5053334455")]
[TestCase("0 505 123 45 67", ExpectedResult = "5051234567")]
[TestCase("0-312-123-45-67", ExpectedResult = "3121234567")]
public string TelefonNoDuzenleTest(string metin);
```

Description

This is TelefonNoDuzenleTest, a member of class MetinIslemleriTest.

Body Source

```
1: [Test]
2: [TestCase("0(505)333-44-55", ExpectedResult = "5053334455")]
3: [TestCase("0 505 123 45 67", ExpectedResult = "5051234567")]
4: [TestCase("0-312-123-45-67", ExpectedResult = "3121234567")]
5: public string TelefonNoDuzenleTest(string metin)
6: {
7:     return MetinIslemleri.TelefonNoDuzenle(metin);
8: }
```

1.1.3.1.1.1.9 MetinIslemleriTest.TurkceKarakterleriDuzeltTest Method

C#

```
[Test]
[TestCase("fistikçi sahap", ExpectedResult = "fistikci sahap")]
[TestCase("ugulu", ExpectedResult = "ugulu")]
[TestCase("ögrenci", ExpectedResult = "ogrenci")]
public string TurkceKarakterleriDuzeltTest(string metin);
```

Description

This is TurkceKarakterleriDuzeltTest, a member of class MetinIslemleriTest.

Body Source

```
1: [Test]
2: [TestCase("fistikçi sahap", ExpectedResult = "fistikci sahap")]
3: [TestCase("ugulu", ExpectedResult = "ugulu")]
4: [TestCase("ögrenci", ExpectedResult = "ogrenci")]
5: public string TurkceKarakterleriDuzeltTest(string metin)
6: {
7:     return MetinIslemleri.TurkceKarakterleriDuzelt(metin);
8: }
```

1.1.3.1.2 SayiIslemleriTest Class

Library (☐ see page 1) kisminda Sayi Islemleri bölümünde Yer alan metotlarin örnek test durumlari ve beklenen sonuçlari ile Test (☐ see page 35) edildikleri siniftir

Class Hierarchy

CSharpHelperMethods.Test.SayiIslemleriTest

C#

```
[TestFixture]
public class SayiIslemleriTest;
```

File

SayiIslemleriTest.cs (☐ see page 117)

SayiIslemleriTest Methods

	Name	Description
⇒◆	SayiHalineCevirTest (☐ see page 40)	This is SayiHalineCevirTest, a member of class SayiIslemleriTest.
⇒◆	SayisalMiTest (☐ see page 40)	This is SayisalMiTest, a member of class SayiIslemleriTest.
⇒◆	SayiyiYaziyaCevirTest (☐ see page 41)	This is SayiyiYaziyaCevirTest, a member of class SayiIslemleriTest.

1.1.3.1.2.1 SayiIslemleriTest Methods

1.1.3.1.2.1.1 SayiIslemleriTest.SayiHalineCevirTest Method

C#

```
[Test]
[TestCase("15_05", ExpectedResult = 15.05)]
[TestCase("456.11", ExpectedResult = 456.11)]
public decimal SayiHalineCevirTest(string metin);
```

Description

This is SayiHalineCevirTest, a member of class SayiIslemleriTest.

Body Source

```
1: [Test]
2: [TestCase("15_05", ExpectedResult = 15.05)]
3: [TestCase("456.11", ExpectedResult = 456.11)]
4: public decimal SayiHalineCevirTest(string metin)
5: {
6:     return SayiIslemleri.SayiHalineCevir(metin);
7: }
```

1.1.3.1.2.1.2 SayiIslemleriTest.SayisalMiTest Method

C#

```
[Test]
[TestCase("15", ExpectedResult = true)]
[TestCase("15123132", ExpectedResult = true)]
[TestCase("3242a", ExpectedResult = false)]
[TestCase("asdada", ExpectedResult = false)]
public bool SayisalMiTest(string metin);
```

Description

This is SayisalMiTest, a member of class SayiIslemleriTest.

Body Source

```
1: [Test]
2: [TestCase("15", ExpectedResult = true)]
3: [TestCase("15123132", ExpectedResult = true)]
4: [TestCase("3242a", ExpectedResult = false)]
5: [TestCase("asdada", ExpectedResult = false)]
6: public bool SayisalMiTest(string metin)
7: {
8:     return SayiIslemleri.SayisalMi(metin);
9: }
```

1.1.3.1.2.1.3 SayiIslemleriTest.SayiyiYaziyaCevirTest Method

C#

```
[Test]
[TestCase(1500, ExpectedResult = "BINBESYÜZ TL SIFIR KR.")]
[TestCase(456789.76, ExpectedResult = "DÖRTYÜZELLIALTIBINYEDIYÜZSEKSENDOKUZ TL YETMISALTI
KR.")]
public string SayiyiYaziyaCevirTest(decimal sayi);
```

Description

This is SayiyiYaziyaCevirTest, a member of class SayiIslemleriTest.

Body Source

```
1: [Test]
2: [TestCase(1500, ExpectedResult = "BINBESYÜZ TL SIFIR KR.")]
3: [TestCase(456789.76, ExpectedResult = "DÖRTYÜZELLIALTIBINYEDIYÜZSEKSENDOKUZ TL
YETMISALTI KR.")]
4: public string SayiyiYaziyaCevirTest(decimal sayi)
5: {
6:     return SayiIslemleri.SayiyiYaziyaCevir(sayi);
7: }
```

1.1.3.1.3 SifrelemeIslemleriTest Class

Library (see page 1) kisminda Sifreleme Islemleri bölümünde Yer alan metotlarin örnek test durumlari ve beklenen sonuçlari ile Test (see page 35) edildikleri siniftir

Class Hierarchy

CSharpHelperMethods.Test.SifrelemeIslemleriTest

C#

```
[TestFixture]
public class SifrelemeIslemleriTest;
```

File

SifrelemeIslemleriTest.cs (see page 121)

SifrelemeIslemleriTest Methods

	Name	Description
⇒♦	HesaplaMD5Test (see page 42)	This is HesaplaMD5Test, a member of class SifrelemeIslemleriTest.
⇒♦	HesaplaSHA256Test (see page 42)	This is HesaplaSHA256Test, a member of class SifrelemeIslemleriTest.
⇒♦	HesaplaSHA512Test (see page 42)	This is HesaplaSHA512Test, a member of class SifrelemeIslemleriTest.
⇒♦	SifreUygunMuTest (see page 43)	This is SifreUygunMuTest, a member of class SifrelemeIslemleriTest.

1.1.3.1.3.1 SifrelemeIslemleriTest Methods

1.1.3.1.3.1.1 SifrelemeIslemleriTest.HesaplaMD5Test Method

C#

```
[Test]
[TestCase("1", ExpectedResult = "C4CA4238A0B923820DCC509A6F75849B")]
[TestCase("serdar", ExpectedResult = "11A37959E906860A34392F586E544FD3")]
public string HesaplaMD5Test(string metin);
```

Description

This is HesaplaMD5Test, a member of class SifrelemeIslemleriTest.

Body Source

```
1: [Test]
2: [TestCase("1", ExpectedResult = "C4CA4238A0B923820DCC509A6F75849B")]
3: [TestCase("serdar", ExpectedResult = "11A37959E906860A34392F586E544FD3")]
4: public string HesaplaMD5Test(string metin)
5: {
6:     return SifrelemeIslemleri.HesaplaMD5(metin);
7: }
```

1.1.3.1.3.1.2 SifrelemeIslemleriTest.HesaplaSHA256Test Method

C#

```
[Test]
[TestCase("deneme", ExpectedResult =
"25B80B3556CA3A15353DD2FD312062FAD27ADCF5A1DE51B75BDADEA1FA8214AB")]
[TestCase("12", ExpectedResult =
"6B51D431DF5D7F141CBECECCF79EDF3DD861C3B4069F0B11661A3EEFACBBA918")]
public string HesaplaSHA256Test(string metin);
```

Description

This is HesaplaSHA256Test, a member of class SifrelemeIslemleriTest.

Body Source

```
1: [Test]
2: [TestCase("deneme", ExpectedResult =
"25B80B3556CA3A15353DD2FD312062FAD27ADCF5A1DE51B75BDADEA1FA8214AB")]
3: [TestCase("12", ExpectedResult =
"6B51D431DF5D7F141CBECECCF79EDF3DD861C3B4069F0B11661A3EEFACBBA918")]
4: public string HesaplaSHA256Test(string metin)
5: {
6:     return SifrelemeIslemleri.HesaplaSHA256(metin);
7: }
```

1.1.3.1.3.1.3 SifrelemeIslemleriTest.HesaplaSHA512Test Method

C#

```
[Test]
[TestCase("34", ExpectedResult =
"E5397F14C44F8DF754617194051DAB1AD38F59F08580406C2EFD59AA4C0F71616713C2ABE76BC503E08F2F5EDA4
863634F6FE99AD39D46C947C09623B91E53CA")]
[TestCase("test", ExpectedResult =
"EE26B0DD4AF7E749AA1A8EE3C10AE9923F618980772E473F8819A5D4940E0DB27AC185F8A0E1D5F84F88BC887FD
67B143732C304CC5FA9AD8E6F57F50028A8FF")]
public string HesaplaSHA512Test(string metin);
```

Description

This is HesaplaSHA512Test, a member of class SifrelemeIslemleriTest.

Body Source

```
1: [Test]
2: [TestCase("34", ExpectedResult =
"E5397F14C44F8DF754617194051DAB1AD38F59F08580406C2EFD59AA4C0F71616713C2ABE76BC503E08F2F5EDA4
```

```
863634F6FE99AD39D46C947C09623B91E53CA")]
3: [TestCase("test", ExpectedResult =
"EE26B0DD4AF7E749AA1A8EE3C10AE9923F618980772E473F8819A5D4940E0DB27AC185F8A0E1D5F84F88BC887FD
67B143732C304CC5FA9AD8E6F57F50028A8FF")]
4: public string HesaplaSHA512Test(string metin)
5: {
6:     return SifrelemeIslemleri.HesaplaSHA512(metin);
7: }
```

1.1.3.1.3.1.4 SifrelemeIslemleriTest.SifreUygunMuTest Method

C#

```
[Test]
[TestCase("1aS_", ExpectedResult = "Parola en az 8 karakter olmalidir.")]
[TestCase("56Serdar", ExpectedResult = "Parola en az bir noktalama isareti içermelidir.")]
[TestCase("54serdar_", ExpectedResult = "Parola en az bir büyük harf içermelidir.")]
[TestCase("12SERADR_", ExpectedResult = "Parola en az bir küçük harf içermelidir.")]
[TestCase("SerdarGul_", ExpectedResult = "Parola en az bir rakam içermelidir.")]
[TestCase("SerdarGul1_", ExpectedResult = "Uygun")]
public string SifreUygunMuTest(string metin);
```

Description

This is SifreUygunMuTest, a member of class SifrelemeIslemleriTest.

Body Source

```
 1: [Test]
 2: [TestCase("1aS_", ExpectedResult = "Parola en az 8 karakter olmalidir.")]
 3: [TestCase("56Serdar", ExpectedResult = "Parola en az bir noktalama isareti
içermelidir.")]
 4: [TestCase("54serdar_", ExpectedResult = "Parola en az bir büyük harf içermelidir.")]
 5: [TestCase("12SERADR_", ExpectedResult = "Parola en az bir küçük harf içermelidir.")]
 6: [TestCase("SerdarGul_", ExpectedResult = "Parola en az bir rakam içermelidir.")]
 7: [TestCase("SerdarGul1_", ExpectedResult = "Uygun")]
 8: public string SifreUygunMuTest(string metin)
 9: {
10:     return SifrelemeIslemleri.SifreUygunMu(metin);
11: }
```

1.1.3.1.4 TarihIslemleriTest Class

Library (⊡ see page 1) kisminda Tarih Islemleri bölümünde Yer alan metotlarin örnek test durumlari ve beklenen sonuçlari ile Test (⊡ see page 35) edildikleri siniftir

Class Hierarchy

CSharpHelperMethods.Test.TarihIslemleriTest

C#

```
[TestFixture]
public class TarihIslemleriTest;
```

File

TarihIslemleriTest.cs (⊡ see page 127)

TarihIslemleriTest Methods

	Name	Description
⇒♦	AyFarkiTest (⊡ see page 44)	This is AyFarkiTest, a member of class TarihIslemleriTest.
⇒♦	KidemTazminatiTarihAraligiHesapla (⊡ see page 44)	This is KidemTazminatiTarihAraligiHesapla, a member of class TarihIslemleriTest.
⇒♦	TarihAraligiHesaplaTest (⊡ see page 44)	This is TarihAraligiHesaplaTest, a member of class TarihIslemleriTest.

⇒◆	YasHesaplaMetinselTest ()	This is YasHesaplaMetinselTest, a member of class TarihIslemleriTest.
⇒◆	YasHesaplaTest ()	This is YasHesaplaTest, a member of class TarihIslemleriTest.

1.1.3.1.4.1 TarihIslemleriTest Methods

1.1.3.1.4.1.1 TarihIslemleriTest.AyFarkiTest Method

C#

```
[Test]
[TestCase(1987, 3, 7, ExpectedResult = 432)]
[TestCase(2022, 1, 7, ExpectedResult = 14)]
public int AyFarkiTest(int yil, int ay, int gun);
```

Description

This is AyFarkiTest, a member of class TarihIslemleriTest.

Body Source

```
1: [Test]
2: [TestCase(1987, 3, 7, ExpectedResult = 432)]
3: [TestCase(2022, 1, 7, ExpectedResult = 14)]
4: public int AyFarkiTest(int yil, int ay, int gun)
5: {
6:     var tarih = new DateTime(yil, ay, gun);
7:     return TarihIslemleri.AyFarki(tarih, DateTime.Now);
8: }
```

1.1.3.1.4.1.2 TarihIslemleriTest.KidemTazminatiTarihAraligiHesapla Method

C#

```
[Test]
[TestCase(1987, 3, 7, ExpectedResult = 12961)]
[TestCase(2022, 1, 7, ExpectedResult = 421)]
public int KidemTazminatiTarihAraligiHesapla(int yil, int ay, int gun);
```

Description

This is KidemTazminatiTarihAraligiHesapla, a member of class TarihIslemleriTest.

Body Source

```
1: [Test]
2: [TestCase(1987, 3, 7, ExpectedResult = 12961)]
3: [TestCase(2022, 1, 7, ExpectedResult = 421)]
4: public int KidemTazminatiTarihAraligiHesapla(int yil, int ay, int gun)
5: {
6:     var tarih = new DateTime(yil, ay, gun);
7:     return TarihIslemleri.KidemTazminatiTarihAraligiHesapla(tarih, DateTime.Now);
8: }
```

1.1.3.1.4.1.3 TarihIslemleriTest.TarihAraligiHesaplaTest Method

C#

```
[Test]
[TestCase(1987, 3, 7, ExpectedResult = "36 Yil, 00 Ay, 00 Gün")]
[TestCase(2022, 1, 7, ExpectedResult = "01 Yil, 02 Ay, 00 Gün")]
public string TarihAraligiHesaplaTest(int yil, int ay, int gun);
```

Description

This is TarihAraligiHesaplaTest, a member of class TarihIslemleriTest.

Body Source

```
1: [Test]
```

```
2: [TestCase(1987, 3, 7, ExpectedResult = "36 Yil, 00 Ay, 00 Gün")]
3: [TestCase(2022, 1, 7, ExpectedResult = "01 Yil, 02 Ay, 00 Gün")]
4: public string TarihAraligiHesaplaTest(int yil, int ay, int gun)
5: {
6:     var tarih = new DateTime(yil, ay, gun);
7:     return TarihIslemleri.TarihAraligiHesapla(tarih, DateTime.Now);
8: }
```

1.1.3.1.4.1.4 TarihIslemleriTest.YasHesaplaMetinselTest Method

C#

```
[Test]
[TestCase(1987, 3, 7, ExpectedResult = "Yas: 36 Yil 0 Ay 0 Gün")]
[TestCase(2022, 1, 7, ExpectedResult = "Yas: 1 Yil 2 Ay 0 Gün")]
public string YasHesaplaMetinselTest(int yil, int ay, int gun);
```

Description

This is YasHesaplaMetinselTest, a member of class TarihIslemleriTest.

Body Source

```
1: [Test]
2: [TestCase(1987, 3, 7, ExpectedResult = "Yas: 36 Yil 0 Ay 0 Gün")]
3: [TestCase(2022, 1, 7, ExpectedResult = "Yas: 1 Yil 2 Ay 0 Gün")]
4: public string YasHesaplaMetinselTest(int yil, int ay, int gun)
5: {
6:     var tarih = new DateTime(yil, ay, gun);
7:     return TarihIslemleri.YasHesaplaMetinsel(tarih);
8: }
```

1.1.3.1.4.1.5 TarihIslemleriTest.YasHesaplaTest Method

C#

```
[Test]
[TestCase(1987, ExpectedResult = 36)]
[TestCase(2023, ExpectedResult = 0)]
public int YasHesaplaTest(int yil);
```

Description

This is YasHesaplaTest, a member of class TarihIslemleriTest.

Body Source

```
1: [Test]
2: [TestCase(1987, ExpectedResult = 36)]
3: [TestCase(2023, ExpectedResult = 0)]
4: public int YasHesaplaTest(int yil)
5: {
6:     var tarih = new DateTime(yil, 1, 1);
7:     return TarihIslemleri.YasHesapla(tarih);
8: }
```

1.1.4 CSharpHelperMethods.UI Namespace

This is namespace CSharpHelperMethods.UI.

Classes

	Name	Description
	FrmAnaSayfa (⊡ see page 46)	This is class CSharpHelperMethods.UI.FrmAnaSayfa.
	Program (⊡ see page 76)	This is class CSharpHelperMethods.UI.Program.

1.1.4.1 Classes

The following table lists classes in this documentation.

Classes

	Name	Description
	FrmAnaSayfa (☑ see page 46)	This is class CSharpHelperMethods.UI.FrmAnaSayfa.
	Program (☑ see page 76)	This is class CSharpHelperMethods.UI.Program.

1.1.4.1.1 FrmAnaSayfa Class

Class Hierarchy
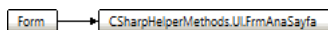
C#

```
public class FrmAnaSayfa : Form;
```

File

FrmAnaSayfa.Designer.cs (☑ see page 93)

Description

This is class CSharpHelperMethods.UI.FrmAnaSayfa.

Methods

	Name	Description
	FrmAnaSayfa (☑ see page 48)	This is FrmAnaSayfa, a member of class FrmAnaSayfa.

FrmAnaSayfa Fields

	Name	Description
	btnAyFarki (☑ see page 48)	This is btnAyFarki, a member of class FrmAnaSayfa.
	btnEPostaDogrula (☑ see page 48)	This is btnEPostaDogrula, a member of class FrmAnaSayfa.
	btnEPostaKontrol (☑ see page 48)	This is btnEPostaKontrol, a member of class FrmAnaSayfa.
	btnIbanDogrula (☑ see page 49)	This is btnIbanDogrula, a member of class FrmAnaSayfa.
	btnIlkHarfiBuyukYap (☑ see page 49)	This is btnIlkHarfiBuyukYap, a member of class FrmAnaSayfa.
	btnIlkHarfleriBuyukYap (☑ see page 49)	This is btnIlkHarfleriBuyukYap, a member of class FrmAnaSayfa.
	btnMd5Hash (☑ see page 49)	This is btnMd5Hash, a member of class FrmAnaSayfa.
	btnMetinSifrele (☑ see page 49)	This is btnMetinSifrele, a member of class FrmAnaSayfa.
	btnMetneDonustur (☑ see page 49)	This is btnMetneDonustur, a member of class FrmAnaSayfa.
	btnOnayKoduOlustur (☑ see page 49)	This is btnOnayKoduOlustur, a member of class FrmAnaSayfa.
	btnRastgeleSifreUret (☑ see page 50)	This is btnRastgeleSifreUret, a member of class FrmAnaSayfa.
	btnSayisalMi (☑ see page 50)	This is btnSayisalMi, a member of class FrmAnaSayfa.
	btnSha256Hash (☑ see page 50)	This is btnSha256Hash, a member of class FrmAnaSayfa.
	btnSha512Hash (☑ see page 50)	This is btnSha512Hash, a member of class FrmAnaSayfa.
	btnSifreUygunMu (☑ see page 50)	This is btnSifreUygunMu, a member of class FrmAnaSayfa.
	btnTarihAraligi (☑ see page 50)	This is btnTarihAraligi, a member of class FrmAnaSayfa.
	btnTcKimlikDogrula (☑ see page 50)	This is btnTcKimlikDogrula, a member of class FrmAnaSayfa.

🔹	btnTelefonNoDuzenle (🔲 see page 51)	This is btnTelefonNoDuzenle, a member of class FrmAnaSayfa.
🔹	btnTurkceKarakterDuzelt (🔲 see page 51)	This is btnTurkceKarakterDuzelt, a member of class FrmAnaSayfa.
🔹	btnYasHesapla (🔲 see page 51)	This is btnYasHesapla, a member of class FrmAnaSayfa.
🔹	btnYasHesaplaMetinsel (🔲 see page 51)	This is btnYasHesaplaMetinsel, a member of class FrmAnaSayfa.
🔹	cmbIslemTuru (🔲 see page 51)	This is cmbIslemTuru, a member of class FrmAnaSayfa.
🔹	components (🔲 see page 51)	Required designer variable.
🔹	dtpBaslangicTarihi (🔲 see page 51)	This is dtpBaslangicTarihi, a member of class FrmAnaSayfa.
🔹	dtpBitisTarihi (🔲 see page 52)	This is dtpBitisTarihi, a member of class FrmAnaSayfa.
🔹	label1 (🔲 see page 52)	This is label1, a member of class FrmAnaSayfa.
🔹	label10 (🔲 see page 52)	This is label10, a member of class FrmAnaSayfa.
🔹	label11 (🔲 see page 52)	This is label11, a member of class FrmAnaSayfa.
🔹	label12 (🔲 see page 52)	This is label12, a member of class FrmAnaSayfa.
🔹	label13 (🔲 see page 52)	This is label13, a member of class FrmAnaSayfa.
🔹	label14 (🔲 see page 52)	This is label14, a member of class FrmAnaSayfa.
🔹	label15 (🔲 see page 53)	This is label15, a member of class FrmAnaSayfa.
🔹	label16 (🔲 see page 53)	This is label16, a member of class FrmAnaSayfa.
🔹	label17 (🔲 see page 53)	This is label17, a member of class FrmAnaSayfa.
🔹	label2 (🔲 see page 53)	This is label2, a member of class FrmAnaSayfa.
🔹	label3 (🔲 see page 53)	This is label3, a member of class FrmAnaSayfa.
🔹	label4 (🔲 see page 53)	This is label4, a member of class FrmAnaSayfa.
🔹	label5 (🔲 see page 53)	This is label5, a member of class FrmAnaSayfa.
🔹	label6 (🔲 see page 54)	This is label6, a member of class FrmAnaSayfa.
🔹	label7 (🔲 see page 54)	This is label7, a member of class FrmAnaSayfa.
🔹	label8 (🔲 see page 54)	This is label8, a member of class FrmAnaSayfa.
🔹	label9 (🔲 see page 54)	This is label9, a member of class FrmAnaSayfa.
🔹	lbKisiSonuc (🔲 see page 54)	This is lbKisiSonuc, a member of class FrmAnaSayfa.
🔹	lbMetinSonuc (🔲 see page 54)	This is lbMetinSonuc, a member of class FrmAnaSayfa.
🔹	lbSayiSonuc (🔲 see page 54)	This is lbSayiSonuc, a member of class FrmAnaSayfa.
🔹	lbSifreSonuc (🔲 see page 55)	This is lbSifreSonuc, a member of class FrmAnaSayfa.
🔹	lbSonuc (🔲 see page 55)	This is lbSonuc, a member of class FrmAnaSayfa.
🔹	pnlKisiIslemleri (🔲 see page 55)	This is pnlKisiIslemleri, a member of class FrmAnaSayfa.
🔹	pnlMetinIslemleri (🔲 see page 55)	This is pnlMetinIslemleri, a member of class FrmAnaSayfa.
🔹	pnlSayiIslemleri (🔲 see page 55)	This is pnlSayiIslemleri, a member of class FrmAnaSayfa.
🔹	pnlSifreIslemleri (🔲 see page 55)	This is pnlSifreIslemleri, a member of class FrmAnaSayfa.
🔹	pnlTarihIslemleri (🔲 see page 55)	This is pnlTarihIslemleri, a member of class FrmAnaSayfa.
🔹	txtKisiMetni (🔲 see page 56)	This is txtKisiMetni, a member of class FrmAnaSayfa.
🔹	txtMetin (🔲 see page 56)	This is txtMetin, a member of class FrmAnaSayfa.
🔹	txtSayi (🔲 see page 56)	This is txtSayi, a member of class FrmAnaSayfa.
🔹	txtSifrelenecekMetin (🔲 see page 56)	This is txtSifrelenecekMetin, a member of class FrmAnaSayfa.

FrmAnaSayfa Methods

	Name	Description
	cmbIslemTuru_SelectedIndexChanged (see page 56)	Islem Türü Combobox'inda seçilen degere göre Ekrandaki panellerden ilgili olan haricinde hepsinin erisilemez hale getirilmesi Eger bir sey seçilmemisse, tamaminin erisilemez hale getirilmesi için Gerekli islemlerin gerçeklestirildigi metottur.
	Dispose (see page 65)	Clean up any resources being used.
	FrmAnaSayfa_Load (see page 65)	Form Ilk Yüklendigi zaman ekrandaki bütün panellerin erisilemez hale getirilmesi saglaniyor Bu metotta, çünkü hangi panelin aktifleseceginin Combobox üzerinden seçilmesi lazim
	InitializeComponent (see page 66)	Required method for Designer support - do not modify the contents of this method with the code editor.

1.1.4.1.1.1 FrmAnaSayfa.FrmAnaSayfa Constructor

C#

```
public FrmAnaSayfa();
```

Description

This is FrmAnaSayfa, a member of class FrmAnaSayfa.

Body Source

```
1: public FrmAnaSayfa()
2: {
3:     InitializeComponent();
4: }
```

1.1.4.1.1.2 FrmAnaSayfa Fields

1.1.4.1.1.2.1 FrmAnaSayfa.btnAyFarki Field

C#

```
private System.Windows.Forms.Button btnAyFarki;
```

Description

This is btnAyFarki, a member of class FrmAnaSayfa.

1.1.4.1.1.2.2 FrmAnaSayfa.btnEPostaDogrula Field

C#

```
private System.Windows.Forms.Button btnEPostaDogrula;
```

Description

This is btnEPostaDogrula, a member of class FrmAnaSayfa.

1.1.4.1.1.2.3 FrmAnaSayfa.btnEPostaKontrol Field

C#

```
private System.Windows.Forms.Button btnEPostaKontrol;
```

Description

This is btnEPostaKontrol, a member of class FrmAnaSayfa.

1.1.4.1.1.2.4 FrmAnaSayfa.btnIbanDogrula Field

C#

```
private System.Windows.Forms.Button btnIbanDogrula;
```

Description

This is btnIbanDogrula, a member of class FrmAnaSayfa.

1.1.4.1.1.2.5 FrmAnaSayfa.btnIlkHarfiBuyukYap Field

C#

```
private System.Windows.Forms.Button btnIlkHarfiBuyukYap;
```

Description

This is btnIlkHarfiBuyukYap, a member of class FrmAnaSayfa.

1.1.4.1.1.2.6 FrmAnaSayfa.btnIlkHarfleriBuyukYap Field

C#

```
private System.Windows.Forms.Button btnIlkHarfleriBuyukYap;
```

Description

This is btnIlkHarfleriBuyukYap, a member of class FrmAnaSayfa.

1.1.4.1.1.2.7 FrmAnaSayfa.btnMd5Hash Field

C#

```
private System.Windows.Forms.Button btnMd5Hash;
```

Description

This is btnMd5Hash, a member of class FrmAnaSayfa.

1.1.4.1.1.2.8 FrmAnaSayfa.btnMetinSifrele Field

C#

```
private System.Windows.Forms.Button btnMetinSifrele;
```

Description

This is btnMetinSifrele, a member of class FrmAnaSayfa.

1.1.4.1.1.2.9 FrmAnaSayfa.btnMetneDonustur Field

C#

```
private System.Windows.Forms.Button btnMetneDonustur;
```

Description

This is btnMetneDonustur, a member of class FrmAnaSayfa.

1.1.4.1.1.2.10 FrmAnaSayfa.btnOnayKoduOlustur Field

C#

```
private System.Windows.Forms.Button btnOnayKoduOlustur;
```

Description

This is btnOnayKoduOlustur, a member of class FrmAnaSayfa.

1.1.4.1.1.2.11 **FrmAnaSayfa.btnRastgeleSifreUret Field**

C#

```
private System.Windows.Forms.Button btnRastgeleSifreUret;
```

Description

This is btnRastgeleSifreUret, a member of class FrmAnaSayfa.

1.1.4.1.1.2.12 **FrmAnaSayfa.btnSayisalMi Field**

C#

```
private System.Windows.Forms.Button btnSayisalMi;
```

Description

This is btnSayisalMi, a member of class FrmAnaSayfa.

1.1.4.1.1.2.13 **FrmAnaSayfa.btnSha256Hash Field**

C#

```
private System.Windows.Forms.Button btnSha256Hash;
```

Description

This is btnSha256Hash, a member of class FrmAnaSayfa.

1.1.4.1.1.2.14 **FrmAnaSayfa.btnSha512Hash Field**

C#

```
private System.Windows.Forms.Button btnSha512Hash;
```

Description

This is btnSha512Hash, a member of class FrmAnaSayfa.

1.1.4.1.1.2.15 **FrmAnaSayfa.btnSifreUygunMu Field**

C#

```
private System.Windows.Forms.Button btnSifreUygunMu;
```

Description

This is btnSifreUygunMu, a member of class FrmAnaSayfa.

1.1.4.1.1.2.16 **FrmAnaSayfa.btnTarihAraligi Field**

C#

```
private System.Windows.Forms.Button btnTarihAraligi;
```

Description

This is btnTarihAraligi, a member of class FrmAnaSayfa.

1.1.4.1.1.2.17 **FrmAnaSayfa.btnTcKimlikDogrula Field**

C#

```
private System.Windows.Forms.Button btnTcKimlikDogrula;
```

Description

This is btnTcKimlikDogrula, a member of class FrmAnaSayfa.

1.1.4.1.1.2.18 FrmAnaSayfa.btnTelefonNoDuzenle Field

C#

```
private System.Windows.Forms.Button btnTelefonNoDuzenle;
```

Description

This is btnTelefonNoDuzenle, a member of class FrmAnaSayfa.

1.1.4.1.1.2.19 FrmAnaSayfa.btnTurkceKarakterDuzelt Field

C#

```
private System.Windows.Forms.Button btnTurkceKarakterDuzelt;
```

Description

This is btnTurkceKarakterDuzelt, a member of class FrmAnaSayfa.

1.1.4.1.1.2.20 FrmAnaSayfa.btnYasHesapla Field

C#

```
private System.Windows.Forms.Button btnYasHesapla;
```

Description

This is btnYasHesapla, a member of class FrmAnaSayfa.

1.1.4.1.1.2.21 FrmAnaSayfa.btnYasHesaplaMetinsel Field

C#

```
private System.Windows.Forms.Button btnYasHesaplaMetinsel;
```

Description

This is btnYasHesaplaMetinsel, a member of class FrmAnaSayfa.

1.1.4.1.1.2.22 FrmAnaSayfa.cmbIslemTuru Field

C#

```
private System.Windows.Forms.ComboBox cmbIslemTuru;
```

Description

This is cmbIslemTuru, a member of class FrmAnaSayfa.

1.1.4.1.1.2.23 FrmAnaSayfa.components Field

Required designer variable.

C#

```
private System.ComponentModel.IContainer components = null;
```

1.1.4.1.1.2.24 FrmAnaSayfa.dtpBaslangicTarihi Field

C#

```
private System.Windows.Forms.DateTimePicker dtpBaslangicTarihi;
```

Description

This is dtpBaslangicTarihi, a member of class FrmAnaSayfa.

1.1.4.1.1.2.25 FrmAnaSayfa.dtpBitisTarihi Field

C#

```
private System.Windows.Forms.DateTimePicker dtpBitisTarihi;
```

Description

This is dtpBitisTarihi, a member of class FrmAnaSayfa.

1.1.4.1.1.2.26 FrmAnaSayfa.label1 Field

C#

```
private System.Windows.Forms.Label label1;
```

Description

This is label1, a member of class FrmAnaSayfa.

1.1.4.1.1.2.27 FrmAnaSayfa.label10 Field

C#

```
private System.Windows.Forms.Label label10;
```

Description

This is label10, a member of class FrmAnaSayfa.

1.1.4.1.1.2.28 FrmAnaSayfa.label11 Field

C#

```
private System.Windows.Forms.Label label11;
```

Description

This is label11, a member of class FrmAnaSayfa.

1.1.4.1.1.2.29 FrmAnaSayfa.label12 Field

C#

```
private System.Windows.Forms.Label label12;
```

Description

This is label12, a member of class FrmAnaSayfa.

1.1.4.1.1.2.30 FrmAnaSayfa.label13 Field

C#

```
private System.Windows.Forms.Label label13;
```

Description

This is label13, a member of class FrmAnaSayfa.

1.1.4.1.1.2.31 FrmAnaSayfa.label14 Field

C#

```
private System.Windows.Forms.Label label14;
```

Description

This is label14, a member of class FrmAnaSayfa.

1.1.4.1.1.2.32 **FrmAnaSayfa.label15 Field**

C#

```
private System.Windows.Forms.Label label15;
```

Description

This is label15, a member of class FrmAnaSayfa.

1.1.4.1.1.2.33 **FrmAnaSayfa.label16 Field**

C#

```
private System.Windows.Forms.Label label16;
```

Description

This is label16, a member of class FrmAnaSayfa.

1.1.4.1.1.2.34 **FrmAnaSayfa.label17 Field**

C#

```
private System.Windows.Forms.Label label17;
```

Description

This is label17, a member of class FrmAnaSayfa.

1.1.4.1.1.2.35 **FrmAnaSayfa.label2 Field**

C#

```
private System.Windows.Forms.Label label2;
```

Description

This is label2, a member of class FrmAnaSayfa.

1.1.4.1.1.2.36 **FrmAnaSayfa.label3 Field**

C#

```
private System.Windows.Forms.Label label3;
```

Description

This is label3, a member of class FrmAnaSayfa.

1.1.4.1.1.2.37 **FrmAnaSayfa.label4 Field**

C#

```
private System.Windows.Forms.Label label4;
```

Description

This is label4, a member of class FrmAnaSayfa.

1.1.4.1.1.2.38 **FrmAnaSayfa.label5 Field**

C#

```
private System.Windows.Forms.Label label5;
```

Description

This is label5, a member of class FrmAnaSayfa.

1.1.4.1.1.2.39 **FrmAnaSayfa.label6 Field**

C#

```
private System.Windows.Forms.Label label6;
```

Description

This is label6, a member of class FrmAnaSayfa.

1.1.4.1.1.2.40 **FrmAnaSayfa.label7 Field**

C#

```
private System.Windows.Forms.Label label7;
```

Description

This is label7, a member of class FrmAnaSayfa.

1.1.4.1.1.2.41 **FrmAnaSayfa.label8 Field**

C#

```
private System.Windows.Forms.Label label8;
```

Description

This is label8, a member of class FrmAnaSayfa.

1.1.4.1.1.2.42 **FrmAnaSayfa.label9 Field**

C#

```
private System.Windows.Forms.Label label9;
```

Description

This is label9, a member of class FrmAnaSayfa.

1.1.4.1.1.2.43 **FrmAnaSayfa.lbKisiSonuc Field**

C#

```
private System.Windows.Forms.ListBox lbKisiSonuc;
```

Description

This is lbKisiSonuc, a member of class FrmAnaSayfa.

1.1.4.1.1.2.44 **FrmAnaSayfa.lbMetinSonuc Field**

C#

```
private System.Windows.Forms.ListBox lbMetinSonuc;
```

Description

This is lbMetinSonuc, a member of class FrmAnaSayfa.

1.1.4.1.1.2.45 **FrmAnaSayfa.lbSayiSonuc Field**

C#

```
private System.Windows.Forms.ListBox lbSayiSonuc;
```

Description

This is lbSayiSonuc, a member of class FrmAnaSayfa.

1.1.4.1.1.2.46 **FrmAnaSayfa.lbSifreSonuc Field**

C#

```
private System.Windows.Forms.ListBox lbSifreSonuc;
```

Description

This is lbSifreSonuc, a member of class FrmAnaSayfa.

1.1.4.1.1.2.47 **FrmAnaSayfa.lbSonuc Field**

C#

```
private System.Windows.Forms.ListBox lbSonuc;
```

Description

This is lbSonuc, a member of class FrmAnaSayfa.

1.1.4.1.1.2.48 **FrmAnaSayfa.pnlKisiIslemleri Field**

C#

```
private System.Windows.Forms.Panel pnlKisiIslemleri;
```

Description

This is pnlKisiIslemleri, a member of class FrmAnaSayfa.

1.1.4.1.1.2.49 **FrmAnaSayfa.pnlMetinIslemleri Field**

C#

```
private System.Windows.Forms.Panel pnlMetinIslemleri;
```

Description

This is pnlMetinIslemleri, a member of class FrmAnaSayfa.

1.1.4.1.1.2.50 **FrmAnaSayfa.pnlSayiIslemleri Field**

C#

```
private System.Windows.Forms.Panel pnlSayiIslemleri;
```

Description

This is pnlSayiIslemleri, a member of class FrmAnaSayfa.

1.1.4.1.1.2.51 **FrmAnaSayfa.pnlSifreIslemleri Field**

C#

```
private System.Windows.Forms.Panel pnlSifreIslemleri;
```

Description

This is pnlSifreIslemleri, a member of class FrmAnaSayfa.

1.1.4.1.1.2.52 **FrmAnaSayfa.pnlTarihIslemleri Field**

C#

```
private System.Windows.Forms.Panel pnlTarihIslemleri;
```

Description

This is pnlTarihIslemleri, a member of class FrmAnaSayfa.

1.1.4.1.1.2.53 **FrmAnaSayfa.txtKisiMetni Field**

C#

```
private System.Windows.Forms.TextBox txtKisiMetni;
```

Description

This is txtKisiMetni, a member of class FrmAnaSayfa.

1.1.4.1.1.2.54 **FrmAnaSayfa.txtMetin Field**

C#

```
private System.Windows.Forms.TextBox txtMetin;
```

Description

This is txtMetin, a member of class FrmAnaSayfa.

1.1.4.1.1.2.55 **FrmAnaSayfa.txtSayi Field**

C#

```
private System.Windows.Forms.TextBox txtSayi;
```

Description

This is txtSayi, a member of class FrmAnaSayfa.

1.1.4.1.1.2.56 **FrmAnaSayfa.txtSifrelenecekMetin Field**

C#

```
private System.Windows.Forms.TextBox txtSifrelenecekMetin;
```

Description

This is txtSifrelenecekMetin, a member of class FrmAnaSayfa.

1.1.4.1.1.3 **FrmAnaSayfa Methods**

1.1.4.1.1.3.1 **FrmAnaSayfa.cmbIslemTuru_SelectedIndexChanged Method**

Islem Türü Combobox'inda seçilen degere göre Ekrandaki panellerden ilgili olan haricinde hepsinin erisilemez hale getirilmesi Eger bir sey seçilmemisse, tamaminin erisilemez hale getirilmesi için Gerekli islemlerin gerçeklestirildigi metottur.

C#

```
private void cmbIslemTuru_SelectedIndexChanged(object sender, EventArgs e);
```

Parameters

Parameters	Description
object sender	The sender info (For example Main Form)
EventArgs e	Event Arguments

Body Source

```
1: private void cmbIslemTuru_SelectedIndexChanged(object sender, EventArgs e)
2:         {
3:             #region Combobox'tan Seçilen Degere Göre Ilgili Paneli Aktif Hale Getirme
4:             switch (cmbIslemTuru.SelectedIndex)
5:             {
6:                 // Metin Islemleri
7:                 case 0:
8:                     {
```

```
 9:                        PaneliErisilebilirKil("pnlMetinIslemleri");
10:                        break;
11:                    }
12:                // Tarih Islemleri
13:                case 1:
14:                    {
15:                        PaneliErisilebilirKil("pnlTarihIslemleri");
16:                        break;
17:                    }
18:                // Sayi Islemleri
19:                case 2:
20:                    {
21:                        PaneliErisilebilirKil("pnlSayiIslemleri");
22:                        break;
23:                    }
24:                // Sifre Islemleri
25:                case 3:
26:                    {
27:                        PaneliErisilebilirKil("pnlSifreIslemleri");
28:                        break;
29:                    }
30:                // Kisi Islemleri
31:                case 4:
32:                    {
33:                        PaneliErisilebilirKil("pnlKisiIslemleri");
34:                        break;
35:                    }
36:                // Bunlar haricinde herhangi bir sey
37:                default:
38:                    {
39:                        PaneliErisilebilirKil("Hicbiri");
40:                        break;
41:                    }
42:            }
43:            #endregion
44:        }
45:
46:        /// <summary>
47:        /// Parametre olarak verilen panelin aktif hale getirilmesi
48:        /// Diger panellerin deaktif edilmesi için hazirlanmis metottur
49:        /// </summary>
50:        /// <param name="panelAdi">Panel Adi Bilgisi</param>
51:        private void PaneliErisilebilirKil(string panelAdi)
52:        {
53:            foreach (Control formElement in Controls)
54:            {
55:                if (formElement.Name == panelAdi)
56:                    formElement.Enabled = true;
57:                else if (formElement.Name.StartsWith("pnl"))
58:                    formElement.Enabled = false;
59:            }
60:        }
61:
62:        /// <summary>
63:        /// Tüm Panelleri gezerek panellerin içerisindeki listboxlarin
64:        /// Multiline özelligine kavusmasi için
65:        /// Gerekli islemlerin düzenlendigi metottur.
66:        /// </summary>
67:        private void ListBoxlariCokluSatirliYap()
68:        {
69:            foreach (Control formElement in Controls)
70:            {
71:                if (formElement.GetType() != typeof(Panel)) continue;
72:                var listBoxList = formElement.Controls.OfType<ListBox>().ToList();
73:                foreach (var listBox in listBoxList)
74:                {
75:                    listBox.DrawMode = DrawMode.OwnerDrawVariable;
76:                    listBox.MeasureItem += listBox_MeasureItem;
77:                    listBox.DrawItem += listBox_DrawItem;
```

```
 78:                    }
 79:                }
 80:            }
 81:
 82:            /// <summary>
 83:            /// This method is used to meaure the length of the item in listbox and make
it multi line
 84:            /// without using scrollbar and let the user see more efficiently
 85:            /// </summary>
 86:            /// <param name="sender">The sender info (For example Main Form)</param>
 87:            /// <param name="e">Event Arguments</param>
 88:            private static void listBox_MeasureItem(object sender, MeasureItemEventArgs e)
 89:            {
 90:                if (!(sender is ListBox listBox)) return;
 91:                e.ItemHeight = (int) e.Graphics
 92:                    .MeasureString(listBox.Items[e.Index].ToString(), listBox.Font,
listBox.Width).Height;
 93:            }
 94:
 95:            /// <summary>
 96:            /// This method is used to meaure the length of the item in listbox and make
it multi line
 97:            /// without using scrollbar and let the user see more efficiently
 98:            /// </summary>
 99:            /// <param name="sender">The sender info (For example Main Form)</param>
100:            /// <param name="e">Event Arguments</param>
101:            private static void listBox_DrawItem(object sender, DrawItemEventArgs e)
102:            {
103:                if (!(sender is ListBox listBox)) return;
104:                e.DrawBackground();
105:                e.DrawFocusRectangle();
106:                e.Graphics.DrawString(listBox.Items[e.Index].ToString(), e.Font, new
SolidBrush(e.ForeColor), e.Bounds);
107:            }
108:
109:            #endregion
110:
111:            #region Tarih Islemleri
112:
113:            /// <summary>
114:            /// Paneldeki Baslangiç Tarihi seçili ve tarih girilmisse,
115:            /// o tarihten bugüne kaç yas oldugunun hesaplandigi metottur.
116:            /// Ilgili butona tiklandigi zaman çalisir
117:            /// </summary>
118:            /// <param name="sender">The sender info (For example Main Form)</param>
119:            /// <param name="e">Event Arguments</param>
120:            private void btnYasHesapla_Click(object sender, EventArgs e)
121:            {
122:                if (!dtpBaslangicTarihi.Checked)
123:                    MessageBox.Show(Sabitler.BaslangicTarihiGirilmedi);
124:                else
125:                {
126:                    lbSonuc.Items.Clear();
127:                    var sb = new StringBuilder();
128:                    sb.Append("Dogum tarihi seçilen kisi ");
129:                    sb.Append(TarihIslemleri.YasHesapla(dtpBaslangicTarihi.Value));
130:                    sb.Append(" yasinin içindedir");
131:                    lbSonuc.Items.Add(sb.ToString());
132:                }
133:            }
134:
135:            /// <summary>
136:            /// Paneldeki Baslangiç Tarihi seçili ve tarih girilmisse,
137:            /// o tarihten bugüne kaç yas oldugunun yil, ay ve gün
138:            /// olarak metinsel bir sekilde hesaplandigi metottur.
139:            /// Ilgili butona tiklandigi zaman çalisir
140:            /// </summary>
141:            /// <param name="sender">The sender info (For example Main Form)</param>
142:            /// <param name="e">Event Arguments</param>
```

```
143:            private void btnYasHesaplaMetinsel_Click(object sender, EventArgs e)
144:            {
145:                if (!dtpBaslangicTarihi.Checked)
146:                    MessageBox.Show(Sabitler.BaslangicTarihiGirilmedi);
147:                else
148:                {
149:                    lbSonuc.Items.Clear();
150:                    var sb = new StringBuilder();
151:                    sb.Append(TarihIslemleri.YasHesaplaMetinsel(dtpBaslangicTarihi.Value));
152:                    lbSonuc.Items.Add(sb.ToString());
153:                }
154:            }
155:
156:            /// <summary>
157:            /// Paneldeki Baslangiç Tarihi ve Bitis Tarihi alanlari seçili ve tarih
girilmisse,
158:            /// Söz konusu tarihler arasinda kaç yil, kaç ay ve gün oldugunun bilgisinin
hazirlandigi metottur.
159:            /// Ilgili butona tiklandigi zaman çalisir
160:            /// </summary>
161:            /// <param name="sender">The sender info (For example Main Form)</param>
162:            /// <param name="e">Event Arguments</param>
163:            private void btnTarihAraligi_Click(object sender, EventArgs e)
164:            {
165:                if (!dtpBaslangicTarihi.Checked || !dtpBitisTarihi.Checked)
166:                    MessageBox.Show(Sabitler.BaslangicVeyaBitisTarihiGirilmedi);
167:                else if (dtpBaslangicTarihi.Value.Date > dtpBitisTarihi.Value.Date)
168:                    MessageBox.Show(Sabitler.BaslangicTarihiBuyuk);
169:                else
170:                {
171:                    lbSonuc.Items.Clear();
172:                    var sb = new StringBuilder();
173:                    sb.Append("Iki tarih arasi: ");
174:
sb.Append(TarihIslemleri.TarihAraligiHesapla(dtpBaslangicTarihi.Value.Date,
dtpBitisTarihi.Value.Date));
175:                    lbSonuc.Items.Add(sb.ToString());
176:                }
177:            }
178:
179:            /// <summary>
180:            /// Paneldeki Baslangiç Tarihi ve Bitis Tarihi alanlari seçili ve tarih
girilmisse,
181:            /// Söz konusu tarihler arasinda kaç ay oldugunun bilgisinin hazirlandigi
metottur.
182:            /// Ilgili butona tiklandigi zaman çalisir
183:            /// </summary>
184:            /// <param name="sender">The sender info (For example Main Form)</param>
185:            /// <param name="e">Event Arguments</param>
186:            private void btnAyFarki_Click(object sender, EventArgs e)
187:            {
188:                if (!dtpBaslangicTarihi.Checked || !dtpBitisTarihi.Checked)
189:                    MessageBox.Show(Sabitler.BaslangicVeyaBitisTarihiGirilmedi);
190:                else if (dtpBaslangicTarihi.Value.Date > dtpBitisTarihi.Value.Date)
191:                    MessageBox.Show(Sabitler.BaslangicTarihiBuyuk);
192:                else
193:                {
194:                    lbSonuc.Items.Clear();
195:                    var sb = new StringBuilder();
196:                    sb.Append("Ay Farki: ");
197:                    sb.Append(TarihIslemleri.AyFarki(dtpBaslangicTarihi.Value.Date,
dtpBitisTarihi.Value.Date));
198:                    lbSonuc.Items.Add(sb.ToString());
199:                }
200:            }
201:            #endregion
202:
203:            #region Sifre Islemleri
204:            /// <summary>
```

```
205:             /// Paneldeki metin kutusu dolu ise
206:             /// Girilen metnin MD5 algoritmasi ile sifrelenmis halini
207:             /// Listbox üzerine yazdiran metottur.
208:             /// </summary>
209:             /// <param name="sender">The sender info (For example Main Form)</param>
210:             /// <param name="e">Event Arguments</param>
211:             private void btnMd5Hash_Click(object sender, EventArgs e)
212:             {
213:                 if (string.IsNullOrEmpty(txtSifrelenecekMetin.Text.Trim()))
214:                     MessageBox.Show(Sabitler.SifrelenecekMetinGirilmedi);
215:                 else
216:                 {
217:                     lbSifreSonuc.Items.Clear();
218:                     var sb = new StringBuilder();
219:                     sb.Append("Girilen Metnin MD5 Karsiligi: ");
220:
sb.Append(SifrelemeIslemleri.HesaplaMD5(txtSifrelenecekMetin.Text.Trim()));
221:                     lbSifreSonuc.Items.Add(sb.ToString());
222:                 }
223:             }
224:
225:             /// <summary>
226:             /// Paneldeki metin kutusu dolu ise
227:             /// Girilen metnin SHA 256 algoritmasi ile sifrelenmis halini
228:             /// Listbox üzerine yazdiran metottur.
229:             /// </summary>
230:             /// <param name="sender">The sender info (For example Main Form)</param>
231:             /// <param name="e">Event Arguments</param>
232:             private void btnSha256Hash_Click(object sender, EventArgs e)
233:             {
234:                 if (string.IsNullOrEmpty(txtSifrelenecekMetin.Text.Trim()))
235:                     MessageBox.Show(Sabitler.SifrelenecekMetinGirilmedi);
236:                 else
237:                 {
238:                     lbSifreSonuc.Items.Clear();
239:                     var sb = new StringBuilder();
240:                     sb.Append("Girilen Metnin SHA 256 Karsiligi: ");
241:
sb.Append(SifrelemeIslemleri.HesaplaSHA256(txtSifrelenecekMetin.Text.Trim()));
242:                     lbSifreSonuc.Items.Add(sb.ToString());
243:                 }
244:             }
245:
246:             /// <summary>
247:             /// Paneldeki metin kutusu dolu ise
248:             /// Girilen metnin SHA 512 algoritmasi ile sifrelenmis halini
249:             /// Listbox üzerine yazdiran metottur.
250:             /// </summary>
251:             /// <param name="sender">The sender info (For example Main Form)</param>
252:             /// <param name="e">Event Arguments</param>
253:             private void btnSha512Hash_Click(object sender, EventArgs e)
254:             {
255:                 if (string.IsNullOrEmpty(txtSifrelenecekMetin.Text.Trim()))
256:                     MessageBox.Show(Sabitler.SifrelenecekMetinGirilmedi);
257:                 else
258:                 {
259:                     lbSifreSonuc.Items.Clear();
260:                     var sb = new StringBuilder();
261:                     sb.Append("Girilen Metnin SHA 512 Karsiligi: ");
262:
sb.Append(SifrelemeIslemleri.HesaplaSHA512(txtSifrelenecekMetin.Text.Trim()));
263:                     lbSifreSonuc.Items.Add(sb.ToString());
264:                 }
265:             }
266:
267:             /// <summary>
268:             /// Ilgili butona tiklandiginda rastgele sifre üretilmesini saglayan metottur
269:             /// </summary>
270:             /// <param name="sender">The sender info (For example Main Form)</param>
```

```
271:            /// <param name="e">Event Arguments</param>
272:            private void btnRastgeleSifreUret_Click(object sender, EventArgs e)
273:            {
274:                lbSifreSonuc.Items.Clear();
275:                var sb = new StringBuilder();
276:                sb.Append("Rastgele Üretilen Sifre: ");
277:                sb.Append(SifrelemeIslemleri.RastgeleSifreUret());
278:                lbSifreSonuc.Items.Add(sb.ToString());
279:            }
280:
281:            /// <summary>
282:            /// Paneldeki metin kutusu dolu ise
283:            /// Burada yazilan metnin bir sifre olarak belirli kriterlere uygun olup
olmadiginin
284:            /// Gösterilmesini saglayan bir metottur
285:            /// </summary>
286:            /// <param name="sender">The sender info (For example Main Form)</param>
287:            /// <param name="e">Event Arguments</param>
288:            private void btnSifreUygunMu_Click(object sender, EventArgs e)
289:            {
290:                if (string.IsNullOrEmpty(txtSifrelenecekMetin.Text.Trim()))
291:                    MessageBox.Show(Sabitler.SifreMetniGirilmedi);
292:                else
293:                {
294:                    lbSifreSonuc.Items.Clear();
295:                    var sb = new StringBuilder();
296:                    sb.Append("Girilen Sifrenin Uygunluk Durumu: ");
297:
sb.Append(SifrelemeIslemleri.SifreUygunMu(txtSifrelenecekMetin.Text.Trim()));
298:                    lbSifreSonuc.Items.Add(sb.ToString());
299:                }
300:            }
301:            #endregion
302:
303:            #region Sayi Islemleri
304:            /// <summary>
305:            /// Paneldeki metin kutusuna girilen metnin sayisal bir deger olup olmadigini
306:            /// Kontrol eden ve sonucu gösteren metottur.
307:            /// </summary>
308:            /// <param name="sender">The sender info (For example Main Form)</param>
309:            /// <param name="e">Event Arguments</param>
310:            private void btnSayisalMi_Click(object sender, EventArgs e)
311:            {
312:                if (string.IsNullOrEmpty(txtSayi.Text.Trim()))
313:                    MessageBox.Show(Sabitler.MetinGirilmedi);
314:                else
315:                {
316:                    lbSayiSonuc.Items.Clear();
317:                    var sb = new StringBuilder();
318:                    sb.Append("Girilen Metin Sayi Mi?: ");
319:                    sb.Append(SayiIslemleri.SayisalMi(txtSayi.Text.Trim()) ? "Evet" :
"Hayir");
320:                    lbSayiSonuc.Items.Add(sb.ToString());
321:                }
322:            }
323:
324:            /// <summary>
325:            /// Paneldeki metin kutusuna girilen para birimi cinsinden rakamin
326:            /// Yaziya çevrilmesini saglayan ve sonucu listbox üzerinde gösteren bir
metottur.
327:            /// </summary>
328:            /// <param name="sender">The sender info (For example Main Form)</param>
329:            /// <param name="e">Event Arguments</param>
330:            private void btnMetneDonustur_Click(object sender, EventArgs e)
331:            {
332:                if (string.IsNullOrEmpty(txtSayi.Text.Trim()))
333:                    MessageBox.Show(Sabitler.MetinGirilmedi);
334:                else if (!SayiIslemleri.SayisalMi(txtSayi.Text.Trim().Replace(",",
".").Replace(".", "")))
```

```
335:                              MessageBox.Show(Sabitler.MetinSayisalDegil);
336:                 else
337:                 {
338:                     var metinSadeceSayiKismi = txtSayi.Text.Trim();
339:                     var sayi = SayiIslemleri.SayiHalineCevir(metinSadeceSayiKismi);
340:                     lbSayiSonuc.Items.Clear();
341:                     var sb = new StringBuilder();
342:                     sb.Append("Girilen Sayinin Metin Hali (Parasal): ");
343:                     sb.Append(SayiIslemleri.SayiyiYaziyaCevir(sayi));
344:                     lbSayiSonuc.Items.Add(sb.ToString());
345:                 }
346:         }
347:
348:         /// <summary>
349:         /// Rastgele olacak sekilde 6 haneli kod olusturmayi saglayan bir metottur.
350:         /// </summary>
351:         /// <param name="sender">The sender info (For example Main Form)</param>
352:         /// <param name="e">Event Arguments</param>
353:         private void btnOnayKoduOlustur_Click(object sender, EventArgs e)
354:         {
355:             lbSayiSonuc.Items.Clear();
356:             var sb = new StringBuilder();
357:             sb.Append("Olusturulan 6 haneli onay kodu: ");
358:             sb.Append(SayiIslemleri.AltiHaneliOnayKoduOlustur());
359:             lbSayiSonuc.Items.Add(sb.ToString());
360:         }
361:         #endregion
362:
363:         #region Kisi Islemleri
364:         /// <summary>
365:         /// Paneldeki metin kutusuna girilen TC Kimlik numarasinin
366:         /// Standart TC Kimlik No algoritmasina uygun olup olmadigini kontrol eden bir
metottur.
367:         /// </summary>
368:         /// <param name="sender">The sender info (For example Main Form)</param>
369:         /// <param name="e">Event Arguments</param>
370:         private void btnTcKimlikDogrula_Click(object sender, EventArgs e)
371:         {
372:             if (string.IsNullOrEmpty(txtKisiMetni.Text.Trim()))
373:                 MessageBox.Show(Sabitler.MetinGirilmedi);
374:             else if (!SayiIslemleri.SayisalMi(txtKisiMetni.Text.Trim()))
375:                 MessageBox.Show(Sabitler.MetinSayisalDegil);
376:             else if (txtKisiMetni.Text.Trim().Length != 11)
377:                 MessageBox.Show(Sabitler.MetinTcKimlikNoDegil);
378:             else
379:             {
380:                 lbKisiSonuc.Items.Clear();
381:                 var sb = new StringBuilder();
382:                 sb.Append("TC Kimlik No Dogru Mu?: ");
383:                 sb.Append(KisiIslemleri.DogrulaTcKimlikNo(txtKisiMetni.Text.Trim()) ?
"Evet" : "Hayir");
384:                 lbKisiSonuc.Items.Add(sb.ToString());
385:             }
386:         }
387:
388:         /// <summary>
389:         /// Paneldeki metin kutusuna girilen IBAN Numarasinin
390:         /// Standart IBAN algoritmasina uygun olup olmadigini kontrol eden bir
metottur.
391:         /// </summary>
392:         /// <param name="sender">The sender info (For example Main Form)</param>
393:         /// <param name="e">Event Arguments</param>
394:         private void btnIbanDogrula_Click(object sender, EventArgs e)
395:         {
396:             if (string.IsNullOrEmpty(txtKisiMetni.Text.Trim()))
397:                 MessageBox.Show(Sabitler.MetinGirilmedi);
398:             else if (txtKisiMetni.Text.Trim().Length != 26)
399:                 MessageBox.Show(Sabitler.MetinIbanNoDegil +
txtKisiMetni.Text.Trim().Length);
```

```
400:                 else
401:                 {
402:                     lbKisiSonuc.Items.Clear();
403:                     var sb = new StringBuilder();
404:                     sb.Append("IBAN Dogru Mu?: ");
405:                     sb.Append(KisiIslemleri.DogrulaIBAN(txtKisiMetni.Text.Trim()) ? "Evet"
: "Hayir");
406:                     lbKisiSonuc.Items.Add(sb.ToString());
407:                 }
408:             }
409:
410:             /// <summary>
411:             /// Paneldeki metin kutusuna girilen metnin
412:             /// Geçerli bir e-posta adresi olup olmadigini kontrol eden bir metottur.
413:             /// </summary>
414:             /// <param name="sender">The sender info (For example Main Form)</param>
415:             /// <param name="e">Event Arguments</param>
416:             private void btnEPostaDogrula_Click(object sender, EventArgs e)
417:             {
418:                 if (string.IsNullOrEmpty(txtKisiMetni.Text.Trim()))
419:                     MessageBox.Show(Sabitler.MetinGirilmedi);
420:                 else
421:                 {
422:                     lbKisiSonuc.Items.Clear();
423:                     var sb = new StringBuilder();
424:                     sb.Append("E-Posta Dogru Mu?: ");
425:                     sb.Append(KisiIslemleri.DogrulaEPosta(txtKisiMetni.Text.Trim()) ?
"Evet" : "Hayir");
426:                     lbKisiSonuc.Items.Add(sb.ToString());
427:                 }
428:             }
429:             #endregion
430:
431:             #region Metin Islemleri
432:             /// <summary>
433:             /// Paneldeki metin kutusuna girilen metnin
434:             /// Kelimelerinin ilk harflerini büyük yapmaya yarayan bir metottur.
435:             /// </summary>
436:             /// <param name="sender">The sender info (For example Main Form)</param>
437:             /// <param name="e">Event Arguments</param>
438:             private void btnIlkHarfleriBuyukYap_Click(object sender, EventArgs e)
439:             {
440:                 if (string.IsNullOrEmpty(txtMetin.Text.Trim()))
441:                     MessageBox.Show(Sabitler.MetinGirilmedi);
442:                 else
443:                 {
444:                     lbMetinSonuc.Items.Clear();
445:                     var sb = new StringBuilder();
446:                     sb.Append("Metnin Düzenlenmis Hali: ");
447:                     sb.Append(txtMetin.Text.Trim().MetinIlkKarakterleriBuyukYap());
448:                     lbMetinSonuc.Items.Add(sb.ToString());
449:                 }
450:             }
451:
452:             /// <summary>
453:             /// Paneldeki metin kutusuna girilen metnin
454:             /// Sadece ilk harfini büyük yapmaya yarayan bir metottur.
455:             /// </summary>
456:             /// <param name="sender">The sender info (For example Main Form)</param>
457:             /// <param name="e">Event Arguments</param>
458:             private void btnIlkHarfiBuyukYap_Click(object sender, EventArgs e)
459:             {
460:                 if (string.IsNullOrEmpty(txtMetin.Text.Trim()))
461:                     MessageBox.Show(Sabitler.MetinGirilmedi);
462:                 else
463:                 {
464:                     lbMetinSonuc.Items.Clear();
465:                     var sb = new StringBuilder();
466:                     sb.Append("Metnin Düzenlenmis Hali: ");
```

```
467:
sb.Append(MetinIslemleri.MetinIlkKarakterBuyukDigerleriKucukYap(txtMetin.Text.Trim()));
468:                lbMetinSonuc.Items.Add(sb.ToString());
469:            }
470:        }
471:
472:        /// <summary>
473:        /// Paneldeki metin kutusuna girilen metnin
474:        /// Içerisindeki Türkçe karakterleri Ingilizee hallerine çeviren bir metottur.
475:        /// </summary>
476:        /// <param name="sender">The sender info (For example Main Form)</param>
477:        /// <param name="e">Event Arguments</param>
478:        private void btnTurkceKarakterDuzelt_Click(object sender, EventArgs e)
479:        {
480:            if (string.IsNullOrEmpty(txtMetin.Text.Trim()))
481:                MessageBox.Show(Sabitler.MetinGirilmedi);
482:            else
483:            {
484:                lbMetinSonuc.Items.Clear();
485:                var sb = new StringBuilder();
486:                sb.Append("Metnin Düzenlenmis Hali: ");
487:
sb.Append(MetinIslemleri.TurkceKarakterleriDuzelt(txtMetin.Text.Trim()));
488:                lbMetinSonuc.Items.Add(sb.ToString());
489:            }
490:        }
491:
492:        /// <summary>
493:        /// KVKK durumlarindan dolayi TC Kimlik No ve Vergi No gibi alanlarin
494:        /// Matbu çiktilarda gizlenmesi gibi bir durum söz konusu oldugundan
495:        /// Girilen bu 10 veya 11 karakterlik metinleri sifrelemeye yarayan bir
metottur
496:        /// </summary>
497:        /// <param name="sender">The sender info (For example Main Form)</param>
498:        /// <param name="e">Event Arguments</param>
499:        private void btnMetinSifrele_Click(object sender, EventArgs e)
500:        {
501:            if (string.IsNullOrEmpty(txtMetin.Text.Trim()))
502:                MessageBox.Show(Sabitler.MetinGirilmedi);
503:            else if (!SayiIslemleri.SayisalMi(txtMetin.Text.Trim()))
504:                MessageBox.Show(Sabitler.MetinSayisalDegil);
505:            else if (txtMetin.Text.Trim().Length != 11 && txtMetin.Text.Trim().Length
!= 10)
506:                MessageBox.Show(Sabitler.MetinTcKimlikNoVeyaVergiNoDegil);
507:            else
508:            {
509:                lbMetinSonuc.Items.Clear();
510:                var sb = new StringBuilder();
511:                sb.Append("TC Kimlik No veya Vergi No'nun Sifrelenmis Hali: ");
512:                sb.Append(MetinIslemleri.MetinSifrele(txtMetin.Text.Trim()));
513:                lbMetinSonuc.Items.Add(sb.ToString());
514:            }
515:        }
516:
517:        /// <summary>
518:        /// Ekrandan çesitli karakterler içeren telefon numaralari alindigi zaman
519:        /// Bu telefon numarasini olmasi gerektigi hale çevirmeye yarayan bir metottur.
520:        /// </summary>
521:        /// <param name="sender">The sender info (For example Main Form)</param>
522:        /// <param name="e">Event Arguments</param>
523:        private void btnTelefonNoDuzenle_Click(object sender, EventArgs e)
524:        {
525:            if (string.IsNullOrEmpty(txtMetin.Text.Trim()))
526:                MessageBox.Show(Sabitler.MetinGirilmedi);
527:            else
528:            {
529:                lbMetinSonuc.Items.Clear();
530:                var sb = new StringBuilder();
531:                sb.Append("Telefon Numarasinin Düzenlenmis Hali: ");
```

```
532:                         sb.Append(MetinIslemleri.TelefonNoDuzenle(txtMetin.Text.Trim()));
533:                         lbMetinSonuc.Items.Add(sb.ToString());
534:                 }
535:         }
536:
537:         /// <summary>
538:         /// Ekrandan girilen metnin geçerli bir e-posta adresi olup olmadigini kontrol
eden
539:         /// Sonucu listbox'a yazdiran metottur.
540:         /// </summary>
541:         /// <param name="sender">The sender info (For example Main Form)</param>
542:         /// <param name="e">Event Arguments</param>
543:         private void btnEPostaKontrol_Click(object sender, EventArgs e)
544:         {
545:             if (string.IsNullOrEmpty(txtMetin.Text.Trim()))
546:                 MessageBox.Show(Sabitler.MetinGirilmedi);
547:             else
548:             {
549:                 lbMetinSonuc.Items.Clear();
550:                 var gecerliMi =
MetinIslemleri.MailAdresiGecerliMi(txtMetin.Text.Trim());
551:                 lbMetinSonuc.Items.Add(txtMetin.Text.Trim());
552:                 lbMetinSonuc.Items.Add(gecerliMi ? Sabitler.EPostaAdresiGecerli :
Sabitler.EPostaAdresiGecerliDegil);
553:             }
554:         }
555:         #endregion
556:     }
557: }
```

1.1.4.1.1.3.2 FrmAnaSayfa.Dispose Method

Clean up any resources being used.

C#

```
protected override void Dispose(bool disposing);
```

Parameters

Parameters	Description
bool disposing	true if managed resources should be disposed; otherwise, false.

Body Source

```
1: protected override void Dispose(bool disposing)
2: {
3:     if (disposing && (components != null))
4:     {
5:         components.Dispose();
6:     }
7:     base.Dispose(disposing);
8: }
```

1.1.4.1.1.3.3 FrmAnaSayfa.FrmAnaSayfa_Load Method

Form Ilk Yüklendigi zaman ekrandaki bütün panellerin erisilemez hale getirilmesi saglaniyor Bu metotta, çünkü hangi panelin aktifleseceginin Combobox üzerinden seçilmesi lazim

C#

```
private void FrmAnaSayfa_Load(object sender, EventArgs e);
```

Parameters

Parameters	Description
object sender	The sender info (For example Main Form)
EventArgs e	Event Arguments

Body Source

```
1: private void FrmAnaSayfa_Load(object sender, EventArgs e)
2: {
3:     cmbIslemTuru.DropDownStyle = ComboBoxStyle.DropDownList;
4:     PaneliErisilebilirKil("Hicbiri");
5:     ListBoxlariCokluSatirliYap();
6: }
```

1.1.4.1.1.3.4 FrmAnaSayfa.InitializeComponent Method

Required method for Designer support - do not modify the contents of this method with the code editor.

C#

```
private void InitializeComponent();
```

Body Source

```
 1: private void InitializeComponent()
 2: {
 3:     this.cmbIslemTuru = new System.Windows.Forms.ComboBox();
 4:     this.label1 = new System.Windows.Forms.Label();
 5:     this.pnlTarihIslemleri = new System.Windows.Forms.Panel();
 6:     this.btnAyFarki = new System.Windows.Forms.Button();
 7:     this.label8 = new System.Windows.Forms.Label();
 8:     this.btnTarihAraligi = new System.Windows.Forms.Button();
 9:     this.btnYasHesaplaMetinsel = new System.Windows.Forms.Button();
10:     this.btnYasHesapla = new System.Windows.Forms.Button();
11:     this.label4 = new System.Windows.Forms.Label();
12:     this.lbSonuc = new System.Windows.Forms.ListBox();
13:     this.dtpBitisTarihi = new System.Windows.Forms.DateTimePicker();
14:     this.dtpBaslangicTarihi = new System.Windows.Forms.DateTimePicker();
15:     this.label3 = new System.Windows.Forms.Label();
16:     this.label2 = new System.Windows.Forms.Label();
17:     this.pnlSifreIslemleri = new System.Windows.Forms.Panel();
18:     this.label7 = new System.Windows.Forms.Label();
19:     this.btnSifreUygunMu = new System.Windows.Forms.Button();
20:     this.btnRastgeleSifreUret = new System.Windows.Forms.Button();
21:     this.btnSha512Hash = new System.Windows.Forms.Button();
22:     this.btnSha256Hash = new System.Windows.Forms.Button();
23:     this.label6 = new System.Windows.Forms.Label();
24:     this.lbSifreSonuc = new System.Windows.Forms.ListBox();
25:     this.btnMd5Hash = new System.Windows.Forms.Button();
26:     this.txtSifrelenecekMetin = new System.Windows.Forms.TextBox();
27:     this.label5 = new System.Windows.Forms.Label();
28:     this.pnlSayiIslemleri = new System.Windows.Forms.Panel();
29:     this.label9 = new System.Windows.Forms.Label();
30:     this.btnOnayKoduOlustur = new System.Windows.Forms.Button();
31:     this.btnMetneDonustur = new System.Windows.Forms.Button();
32:     this.label10 = new System.Windows.Forms.Label();
33:     this.lbSayiSonuc = new System.Windows.Forms.ListBox();
34:     this.btnSayisalMi = new System.Windows.Forms.Button();
35:     this.txtSayi = new System.Windows.Forms.TextBox();
36:     this.label11 = new System.Windows.Forms.Label();
37:     this.pnlKisiIslemleri = new System.Windows.Forms.Panel();
38:     this.label12 = new System.Windows.Forms.Label();
39:     this.btnEPostaDogrula = new System.Windows.Forms.Button();
40:     this.btnIbanDogrula = new System.Windows.Forms.Button();
41:     this.label13 = new System.Windows.Forms.Label();
42:     this.lbKisiSonuc = new System.Windows.Forms.ListBox();
43:     this.btnTcKimlikDogrula = new System.Windows.Forms.Button();
44:     this.txtKisiMetni = new System.Windows.Forms.TextBox();
45:     this.label14 = new System.Windows.Forms.Label();
46:     this.pnlMetinIslemleri = new System.Windows.Forms.Panel();
47:     this.btnTelefonNoDuzenle = new System.Windows.Forms.Button();
48:     this.btnMetinSifrele = new System.Windows.Forms.Button();
49:     this.label15 = new System.Windows.Forms.Label();
50:     this.btnTurkceKarakterDuzelt = new System.Windows.Forms.Button();
```

```
 51:        this.btnIlkHarfiBuyukYap = new System.Windows.Forms.Button();
 52:        this.label16 = new System.Windows.Forms.Label();
 53:        this.lbMetinSonuc = new System.Windows.Forms.ListBox();
 54:        this.btnIlkHarfleriBuyukYap = new System.Windows.Forms.Button();
 55:        this.txtMetin = new System.Windows.Forms.TextBox();
 56:        this.label17 = new System.Windows.Forms.Label();
 57:        this.btnEPostaKontrol = new System.Windows.Forms.Button();
 58:        this.pnlTarihIslemleri.SuspendLayout();
 59:        this.pnlSifreIslemleri.SuspendLayout();
 60:        this.pnlSayiIslemleri.SuspendLayout();
 61:        this.pnlKisiIslemleri.SuspendLayout();
 62:        this.pnlMetinIslemleri.SuspendLayout();
 63:        this.SuspendLayout();
 64:        //
 65:        // cmbIslemTuru
 66:        //
 67:        this.cmbIslemTuru.FormattingEnabled = true;
 68:        this.cmbIslemTuru.Items.AddRange(new object[] {"Metin Islemleri", "Tarih
Islemleri", "Sayi Islemleri", "Sifre Islemleri", "Kisi Islemleri"});
 69:        this.cmbIslemTuru.Location = new System.Drawing.Point(176, 16);
 70:        this.cmbIslemTuru.Margin = new System.Windows.Forms.Padding(4, 4, 4, 4);
 71:        this.cmbIslemTuru.Name = "cmbIslemTuru";
 72:        this.cmbIslemTuru.Size = new System.Drawing.Size(265, 24);
 73:        this.cmbIslemTuru.TabIndex = 0;
 74:        this.cmbIslemTuru.SelectedIndexChanged += new
System.EventHandler(this.cmbIslemTuru_SelectedIndexChanged);
 75:        //
 76:        // label1
 77:        //
 78:        this.label1.AutoSize = true;
 79:        this.label1.Location = new System.Drawing.Point(16, 20);
 80:        this.label1.Margin = new System.Windows.Forms.Padding(4, 0, 4, 0);
 81:        this.label1.Name = "label1";
 82:        this.label1.Size = new System.Drawing.Size(123, 17);
 83:        this.label1.TabIndex = 1;
 84:        this.label1.Text = "Islem Türü Seçiniz";
 85:        //
 86:        // pnlTarihIslemleri
 87:        //
 88:        this.pnlTarihIslemleri.BorderStyle = System.Windows.Forms.BorderStyle.FixedSingle;
 89:        this.pnlTarihIslemleri.Controls.Add(this.btnAyFarki);
 90:        this.pnlTarihIslemleri.Controls.Add(this.label8);
 91:        this.pnlTarihIslemleri.Controls.Add(this.btnTarihAraligi);
 92:        this.pnlTarihIslemleri.Controls.Add(this.btnYasHesaplaMetinsel);
 93:        this.pnlTarihIslemleri.Controls.Add(this.btnYasHesapla);
 94:        this.pnlTarihIslemleri.Controls.Add(this.label4);
 95:        this.pnlTarihIslemleri.Controls.Add(this.lbSonuc);
 96:        this.pnlTarihIslemleri.Controls.Add(this.dtpBitisTarihi);
 97:        this.pnlTarihIslemleri.Controls.Add(this.dtpBaslangicTarihi);
 98:        this.pnlTarihIslemleri.Controls.Add(this.label3);
 99:        this.pnlTarihIslemleri.Controls.Add(this.label2);
100:        this.pnlTarihIslemleri.Location = new System.Drawing.Point(20, 69);
101:        this.pnlTarihIslemleri.Margin = new System.Windows.Forms.Padding(4, 4, 4, 4);
102:        this.pnlTarihIslemleri.Name = "pnlTarihIslemleri";
103:        this.pnlTarihIslemleri.Size = new System.Drawing.Size(498, 222);
104:        this.pnlTarihIslemleri.TabIndex = 2;
105:        //
106:        // btnAyFarki
107:        //
108:        this.btnAyFarki.Location = new System.Drawing.Point(381, 94);
109:        this.btnAyFarki.Margin = new System.Windows.Forms.Padding(4, 4, 4, 4);
110:        this.btnAyFarki.Name = "btnAyFarki";
111:        this.btnAyFarki.Size = new System.Drawing.Size(105, 28);
112:        this.btnAyFarki.TabIndex = 12;
113:        this.btnAyFarki.Text = "Ay Farki";
114:        this.btnAyFarki.UseVisualStyleBackColor = true;
115:        this.btnAyFarki.Click += new System.EventHandler(this.btnAyFarki_Click);
116:        //
117:        // label8
```

```
118:        //
119:        this.label8.AutoSize = true;
120:        this.label8.Dock = System.Windows.Forms.DockStyle.Top;
121:        this.label8.Location = new System.Drawing.Point(0, 0);
122:        this.label8.Margin = new System.Windows.Forms.Padding(4, 0, 4, 0);
123:        this.label8.Name = "label8";
124:        this.label8.Size = new System.Drawing.Size(96, 17);
125:        this.label8.TabIndex = 11;
126:        this.label8.Text = "Tarih Islemleri";
127:        //
128:        // btnTarihAraligi
129:        //
130:        this.btnTarihAraligi.Location = new System.Drawing.Point(272, 94);
131:        this.btnTarihAraligi.Margin = new System.Windows.Forms.Padding(4, 4, 4, 4);
132:        this.btnTarihAraligi.Name = "btnTarihAraligi";
133:        this.btnTarihAraligi.Size = new System.Drawing.Size(101, 28);
134:        this.btnTarihAraligi.TabIndex = 10;
135:        this.btnTarihAraligi.Text = "Tarih Araligi";
136:        this.btnTarihAraligi.UseVisualStyleBackColor = true;
137:        this.btnTarihAraligi.Click += new System.EventHandler(this.btnTarihAraligi_Click);
138:        //
139:        // btnYasHesaplaMetinsel
140:        //
141:        this.btnYasHesaplaMetinsel.Location = new System.Drawing.Point(112, 94);
142:        this.btnYasHesaplaMetinsel.Margin = new System.Windows.Forms.Padding(4, 4, 4, 4);
143:        this.btnYasHesaplaMetinsel.Name = "btnYasHesaplaMetinsel";
144:        this.btnYasHesaplaMetinsel.Size = new System.Drawing.Size(151, 28);
145:        this.btnYasHesaplaMetinsel.TabIndex = 9;
146:        this.btnYasHesaplaMetinsel.Text = "Yas Hesapla (Metin)";
147:        this.btnYasHesaplaMetinsel.UseVisualStyleBackColor = true;
148:        this.btnYasHesaplaMetinsel.Click += new
System.EventHandler(this.btnYasHesaplaMetinsel_Click);
149:        //
150:        // btnYasHesapla
151:        //
152:        this.btnYasHesapla.Location = new System.Drawing.Point(4, 94);
153:        this.btnYasHesapla.Margin = new System.Windows.Forms.Padding(4, 4, 4, 4);
154:        this.btnYasHesapla.Name = "btnYasHesapla";
155:        this.btnYasHesapla.Size = new System.Drawing.Size(100, 28);
156:        this.btnYasHesapla.TabIndex = 8;
157:        this.btnYasHesapla.Text = "Yas Hesapla";
158:        this.btnYasHesapla.UseVisualStyleBackColor = true;
159:        this.btnYasHesapla.Click += new System.EventHandler(this.btnYasHesapla_Click);
160:        //
161:        // label4
162:        //
163:        this.label4.AutoSize = true;
164:        this.label4.Location = new System.Drawing.Point(24, 134);
165:        this.label4.Margin = new System.Windows.Forms.Padding(4, 0, 4, 0);
166:        this.label4.Name = "label4";
167:        this.label4.Size = new System.Drawing.Size(48, 17);
168:        this.label4.TabIndex = 7;
169:        this.label4.Text = "Sonuç";
170:        //
171:        // lbSonuc
172:        //
173:        this.lbSonuc.FormattingEnabled = true;
174:        this.lbSonuc.HorizontalScrollbar = true;
175:        this.lbSonuc.ItemHeight = 16;
176:        this.lbSonuc.Location = new System.Drawing.Point(83, 129);
177:        this.lbSonuc.Margin = new System.Windows.Forms.Padding(4, 4, 4, 4);
178:        this.lbSonuc.Name = "lbSonuc";
179:        this.lbSonuc.Size = new System.Drawing.Size(340, 84);
180:        this.lbSonuc.TabIndex = 6;
181:        //
182:        // dtpBitisTarihi
183:        //
184:        this.dtpBitisTarihi.Format = System.Windows.Forms.DateTimePickerFormat.Short;
185:        this.dtpBitisTarihi.Location = new System.Drawing.Point(156, 53);
```

```
186:        this.dtpBitisTarihi.Margin = new System.Windows.Forms.Padding(4, 4, 4, 4);
187:        this.dtpBitisTarihi.Name = "dtpBitisTarihi";
188:        this.dtpBitisTarihi.ShowCheckBox = true;
189:        this.dtpBitisTarihi.Size = new System.Drawing.Size(265, 22);
190:        this.dtpBitisTarihi.TabIndex = 5;
191:        //
192:        // dtpBaslangicTarihi
193:        //
194:        this.dtpBaslangicTarihi.Format = System.Windows.Forms.DateTimePickerFormat.Short;
195:        this.dtpBaslangicTarihi.Location = new System.Drawing.Point(156, 21);
196:        this.dtpBaslangicTarihi.Margin = new System.Windows.Forms.Padding(4, 4, 4, 4);
197:        this.dtpBaslangicTarihi.Name = "dtpBaslangicTarihi";
198:        this.dtpBaslangicTarihi.ShowCheckBox = true;
199:        this.dtpBaslangicTarihi.Size = new System.Drawing.Size(265, 22);
200:        this.dtpBaslangicTarihi.TabIndex = 4;
201:        //
202:        // label3
203:        //
204:        this.label3.AutoSize = true;
205:        this.label3.Location = new System.Drawing.Point(59, 58);
206:        this.label3.Margin = new System.Windows.Forms.Padding(4, 0, 4, 0);
207:        this.label3.Name = "label3";
208:        this.label3.Size = new System.Drawing.Size(74, 17);
209:        this.label3.TabIndex = 3;
210:        this.label3.Text = "Bitis Tarihi";
211:        //
212:        // label2
213:        //
214:        this.label2.AutoSize = true;
215:        this.label2.Location = new System.Drawing.Point(23, 30);
216:        this.label2.Margin = new System.Windows.Forms.Padding(4, 0, 4, 0);
217:        this.label2.Name = "label2";
218:        this.label2.Size = new System.Drawing.Size(109, 17);
219:        this.label2.TabIndex = 2;
220:        this.label2.Text = "Baslangiç Tarihi";
221:        //
222:        // pnlSifreIslemleri
223:        //
224:        this.pnlSifreIslemleri.BorderStyle = System.Windows.Forms.BorderStyle.FixedSingle;
225:        this.pnlSifreIslemleri.Controls.Add(this.label7);
226:        this.pnlSifreIslemleri.Controls.Add(this.btnSifreUygunMu);
227:        this.pnlSifreIslemleri.Controls.Add(this.btnRastgeleSifreUret);
228:        this.pnlSifreIslemleri.Controls.Add(this.btnSha512Hash);
229:        this.pnlSifreIslemleri.Controls.Add(this.btnSha256Hash);
230:        this.pnlSifreIslemleri.Controls.Add(this.label6);
231:        this.pnlSifreIslemleri.Controls.Add(this.lbSifreSonuc);
232:        this.pnlSifreIslemleri.Controls.Add(this.btnMd5Hash);
233:        this.pnlSifreIslemleri.Controls.Add(this.txtSifrelenecekMetin);
234:        this.pnlSifreIslemleri.Controls.Add(this.label5);
235:        this.pnlSifreIslemleri.Location = new System.Drawing.Point(548, 69);
236:        this.pnlSifreIslemleri.Margin = new System.Windows.Forms.Padding(4, 4, 4, 4);
237:        this.pnlSifreIslemleri.Name = "pnlSifreIslemleri";
238:        this.pnlSifreIslemleri.Size = new System.Drawing.Size(502, 222);
239:        this.pnlSifreIslemleri.TabIndex = 3;
240:        //
241:        // label7
242:        //
243:        this.label7.AutoSize = true;
244:        this.label7.Dock = System.Windows.Forms.DockStyle.Top;
245:        this.label7.Location = new System.Drawing.Point(0, 0);
246:        this.label7.Margin = new System.Windows.Forms.Padding(4, 0, 4, 0);
247:        this.label7.Name = "label7";
248:        this.label7.Size = new System.Drawing.Size(92, 17);
249:        this.label7.TabIndex = 18;
250:        this.label7.Text = "Sifre Islemleri";
251:        this.label7.TextAlign = System.Drawing.ContentAlignment.TopCenter;
252:        //
253:        // btnSifreUygunMu
254:        //
```

```
255:        this.btnSifreUygunMu.Location = new System.Drawing.Point(231, 87);
256:        this.btnSifreUygunMu.Margin = new System.Windows.Forms.Padding(4, 4, 4, 4);
257:        this.btnSifreUygunMu.Name = "btnSifreUygunMu";
258:        this.btnSifreUygunMu.Size = new System.Drawing.Size(193, 28);
259:        this.btnSifreUygunMu.TabIndex = 17;
260:        this.btnSifreUygunMu.Text = "Sifre Uygun Mu?";
261:        this.btnSifreUygunMu.UseVisualStyleBackColor = true;
262:        this.btnSifreUygunMu.Click += new System.EventHandler(this.btnSifreUygunMu_Click);
263:        //
264:        // btnRastgeleSifreUret
265:        //
266:        this.btnRastgeleSifreUret.Location = new System.Drawing.Point(28, 87);
267:        this.btnRastgeleSifreUret.Margin = new System.Windows.Forms.Padding(4, 4, 4, 4);
268:        this.btnRastgeleSifreUret.Name = "btnRastgeleSifreUret";
269:        this.btnRastgeleSifreUret.Size = new System.Drawing.Size(193, 28);
270:        this.btnRastgeleSifreUret.TabIndex = 16;
271:        this.btnRastgeleSifreUret.Text = "Rastgele Sifre Üret";
272:        this.btnRastgeleSifreUret.UseVisualStyleBackColor = true;
273:        this.btnRastgeleSifreUret.Click += new
System.EventHandler(this.btnRastgeleSifreUret_Click);
274:        //
275:        // btnSha512Hash
276:        //
277:        this.btnSha512Hash.Location = new System.Drawing.Point(324, 52);
278:        this.btnSha512Hash.Margin = new System.Windows.Forms.Padding(4, 4, 4, 4);
279:        this.btnSha512Hash.Name = "btnSha512Hash";
280:        this.btnSha512Hash.Size = new System.Drawing.Size(100, 28);
281:        this.btnSha512Hash.TabIndex = 15;
282:        this.btnSha512Hash.Text = "SHA 512";
283:        this.btnSha512Hash.UseVisualStyleBackColor = true;
284:        this.btnSha512Hash.Click += new System.EventHandler(this.btnSha512Hash_Click);
285:        //
286:        // btnSha256Hash
287:        //
288:        this.btnSha256Hash.Location = new System.Drawing.Point(175, 52);
289:        this.btnSha256Hash.Margin = new System.Windows.Forms.Padding(4, 4, 4, 4);
290:        this.btnSha256Hash.Name = "btnSha256Hash";
291:        this.btnSha256Hash.Size = new System.Drawing.Size(100, 28);
292:        this.btnSha256Hash.TabIndex = 14;
293:        this.btnSha256Hash.Text = "SHA 256";
294:        this.btnSha256Hash.UseVisualStyleBackColor = true;
295:        this.btnSha256Hash.Click += new System.EventHandler(this.btnSha256Hash_Click);
296:        //
297:        // label6
298:        //
299:        this.label6.AutoSize = true;
300:        this.label6.Location = new System.Drawing.Point(28, 123);
301:        this.label6.Margin = new System.Windows.Forms.Padding(4, 0, 4, 0);
302:        this.label6.Name = "label6";
303:        this.label6.Size = new System.Drawing.Size(48, 17);
304:        this.label6.TabIndex = 13;
305:        this.label6.Text = "Sonuç";
306:        //
307:        // lbSifreSonuc
308:        //
309:        this.lbSifreSonuc.FormattingEnabled = true;
310:        this.lbSifreSonuc.HorizontalScrollbar = true;
311:        this.lbSifreSonuc.ItemHeight = 16;
312:        this.lbSifreSonuc.Location = new System.Drawing.Point(87, 123);
313:        this.lbSifreSonuc.Margin = new System.Windows.Forms.Padding(4, 4, 4, 4);
314:        this.lbSifreSonuc.Name = "lbSifreSonuc";
315:        this.lbSifreSonuc.Size = new System.Drawing.Size(333, 84);
316:        this.lbSifreSonuc.TabIndex = 12;
317:        //
318:        // btnMd5Hash
319:        //
320:        this.btnMd5Hash.Location = new System.Drawing.Point(28, 52);
321:        this.btnMd5Hash.Margin = new System.Windows.Forms.Padding(4, 4, 4, 4);
322:        this.btnMd5Hash.Name = "btnMd5Hash";
```

```
323:        this.btnMd5Hash.Size = new System.Drawing.Size(100, 28);
324:        this.btnMd5Hash.TabIndex = 11;
325:        this.btnMd5Hash.Text = "MD5";
326:        this.btnMd5Hash.UseVisualStyleBackColor = true;
327:        this.btnMd5Hash.Click += new System.EventHandler(this.btnMd5Hash_Click);
328:        //
329:        // txtSifrelenecekMetin
330:        //
331:        this.txtSifrelenecekMetin.Location = new System.Drawing.Point(157, 20);
332:        this.txtSifrelenecekMetin.Margin = new System.Windows.Forms.Padding(4, 4, 4, 4);
333:        this.txtSifrelenecekMetin.Name = "txtSifrelenecekMetin";
334:        this.txtSifrelenecekMetin.Size = new System.Drawing.Size(264, 22);
335:        this.txtSifrelenecekMetin.TabIndex = 4;
336:        //
337:        // label5
338:        //
339:        this.label5.AutoSize = true;
340:        this.label5.Location = new System.Drawing.Point(23, 23);
341:        this.label5.Margin = new System.Windows.Forms.Padding(4, 0, 4, 0);
342:        this.label5.Name = "label5";
343:        this.label5.Size = new System.Drawing.Size(124, 17);
344:        this.label5.TabIndex = 3;
345:        this.label5.Text = "Sifrelenecek Metin";
346:        //
347:        // pnlSayiIslemleri
348:        //
349:        this.pnlSayiIslemleri.BorderStyle = System.Windows.Forms.BorderStyle.FixedSingle;
350:        this.pnlSayiIslemleri.Controls.Add(this.label9);
351:        this.pnlSayiIslemleri.Controls.Add(this.btnOnayKoduOlustur);
352:        this.pnlSayiIslemleri.Controls.Add(this.btnMetneDonustur);
353:        this.pnlSayiIslemleri.Controls.Add(this.label10);
354:        this.pnlSayiIslemleri.Controls.Add(this.lbSayiSonuc);
355:        this.pnlSayiIslemleri.Controls.Add(this.btnSayisalMi);
356:        this.pnlSayiIslemleri.Controls.Add(this.txtSayi);
357:        this.pnlSayiIslemleri.Controls.Add(this.label11);
358:        this.pnlSayiIslemleri.Location = new System.Drawing.Point(20, 316);
359:        this.pnlSayiIslemleri.Margin = new System.Windows.Forms.Padding(4, 4, 4, 4);
360:        this.pnlSayiIslemleri.Name = "pnlSayiIslemleri";
361:        this.pnlSayiIslemleri.Size = new System.Drawing.Size(502, 222);
362:        this.pnlSayiIslemleri.TabIndex = 19;
363:        //
364:        // label9
365:        //
366:        this.label9.AutoSize = true;
367:        this.label9.Dock = System.Windows.Forms.DockStyle.Top;
368:        this.label9.Location = new System.Drawing.Point(0, 0);
369:        this.label9.Margin = new System.Windows.Forms.Padding(4, 0, 4, 0);
370:        this.label9.Name = "label9";
371:        this.label9.Size = new System.Drawing.Size(90, 17);
372:        this.label9.TabIndex = 18;
373:        this.label9.Text = "Sayi Islemleri";
374:        this.label9.TextAlign = System.Drawing.ContentAlignment.TopCenter;
375:        //
376:        // btnOnayKoduOlustur
377:        //
378:        this.btnOnayKoduOlustur.Location = new System.Drawing.Point(324, 52);
379:        this.btnOnayKoduOlustur.Margin = new System.Windows.Forms.Padding(4, 4, 4, 4);
380:        this.btnOnayKoduOlustur.Name = "btnOnayKoduOlustur";
381:        this.btnOnayKoduOlustur.Size = new System.Drawing.Size(100, 28);
382:        this.btnOnayKoduOlustur.TabIndex = 15;
383:        this.btnOnayKoduOlustur.Text = "Onay Kodu";
384:        this.btnOnayKoduOlustur.UseVisualStyleBackColor = true;
385:        this.btnOnayKoduOlustur.Click += new
System.EventHandler(this.btnOnayKoduOlustur_Click);
386:        //
387:        // btnMetneDonustur
388:        //
389:        this.btnMetneDonustur.Location = new System.Drawing.Point(136, 52);
390:        this.btnMetneDonustur.Margin = new System.Windows.Forms.Padding(4, 4, 4, 4);
```

```
391:        this.btnMetneDonustur.Name = "btnMetneDonustur";
392:        this.btnMetneDonustur.Size = new System.Drawing.Size(180, 28);
393:        this.btnMetneDonustur.TabIndex = 14;
394:        this.btnMetneDonustur.Text = "Para Metnine Dönüstür";
395:        this.btnMetneDonustur.UseVisualStyleBackColor = true;
396:        this.btnMetneDonustur.Click += new
System.EventHandler(this.btnMetneDonustur_Click);
397:        //
398:        // label10
399:        //
400:        this.label10.AutoSize = true;
401:        this.label10.Location = new System.Drawing.Point(24, 96);
402:        this.label10.Margin = new System.Windows.Forms.Padding(4, 0, 4, 0);
403:        this.label10.Name = "label10";
404:        this.label10.Size = new System.Drawing.Size(48, 17);
405:        this.label10.TabIndex = 13;
406:        this.label10.Text = "Sonuç";
407:        //
408:        // lbSayiSonuc
409:        //
410:        this.lbSayiSonuc.FormattingEnabled = true;
411:        this.lbSayiSonuc.HorizontalScrollbar = true;
412:        this.lbSayiSonuc.ItemHeight = 16;
413:        this.lbSayiSonuc.Location = new System.Drawing.Point(83, 96);
414:        this.lbSayiSonuc.Margin = new System.Windows.Forms.Padding(4, 4, 4, 4);
415:        this.lbSayiSonuc.Name = "lbSayiSonuc";
416:        this.lbSayiSonuc.Size = new System.Drawing.Size(337, 100);
417:        this.lbSayiSonuc.TabIndex = 12;
418:        //
419:        // btnSayisalMi
420:        //
421:        this.btnSayisalMi.Location = new System.Drawing.Point(28, 52);
422:        this.btnSayisalMi.Margin = new System.Windows.Forms.Padding(4, 4, 4, 4);
423:        this.btnSayisalMi.Name = "btnSayisalMi";
424:        this.btnSayisalMi.Size = new System.Drawing.Size(100, 28);
425:        this.btnSayisalMi.TabIndex = 11;
426:        this.btnSayisalMi.Text = "Sayisal Mi?";
427:        this.btnSayisalMi.UseVisualStyleBackColor = true;
428:        this.btnSayisalMi.Click += new System.EventHandler(this.btnSayisalMi_Click);
429:        //
430:        // txtSayi
431:        //
432:        this.txtSayi.Location = new System.Drawing.Point(157, 20);
433:        this.txtSayi.Margin = new System.Windows.Forms.Padding(4, 4, 4, 4);
434:        this.txtSayi.Name = "txtSayi";
435:        this.txtSayi.Size = new System.Drawing.Size(264, 22);
436:        this.txtSayi.TabIndex = 4;
437:        //
438:        // label11
439:        //
440:        this.label11.AutoSize = true;
441:        this.label11.Location = new System.Drawing.Point(23, 23);
442:        this.label11.Margin = new System.Windows.Forms.Padding(4, 0, 4, 0);
443:        this.label11.Name = "label11";
444:        this.label11.Size = new System.Drawing.Size(124, 17);
445:        this.label11.TabIndex = 3;
446:        this.label11.Text = "Sifrelenecek Metin";
447:        //
448:        // pnlKisiIslemleri
449:        //
450:        this.pnlKisiIslemleri.BorderStyle = System.Windows.Forms.BorderStyle.FixedSingle;
451:        this.pnlKisiIslemleri.Controls.Add(this.label12);
452:        this.pnlKisiIslemleri.Controls.Add(this.btnEPostaDogrula);
453:        this.pnlKisiIslemleri.Controls.Add(this.btnIbanDogrula);
454:        this.pnlKisiIslemleri.Controls.Add(this.label13);
455:        this.pnlKisiIslemleri.Controls.Add(this.lbKisiSonuc);
456:        this.pnlKisiIslemleri.Controls.Add(this.btnTcKimlikDogrula);
457:        this.pnlKisiIslemleri.Controls.Add(this.txtKisiMetni);
458:        this.pnlKisiIslemleri.Controls.Add(this.label14);
```

```
459:        this.pnlKisiIslemleri.Location = new System.Drawing.Point(548, 316);
460:        this.pnlKisiIslemleri.Margin = new System.Windows.Forms.Padding(4, 4, 4, 4);
461:        this.pnlKisiIslemleri.Name = "pnlKisiIslemleri";
462:        this.pnlKisiIslemleri.Size = new System.Drawing.Size(502, 222);
463:        this.pnlKisiIslemleri.TabIndex = 20;
464:        //
465:        // label12
466:        //
467:        this.label12.AutoSize = true;
468:        this.label12.Dock = System.Windows.Forms.DockStyle.Top;
469:        this.label12.Location = new System.Drawing.Point(0, 0);
470:        this.label12.Margin = new System.Windows.Forms.Padding(4, 0, 4, 0);
471:        this.label12.Name = "label12";
472:        this.label12.Size = new System.Drawing.Size(85, 17);
473:        this.label12.TabIndex = 18;
474:        this.label12.Text = "Kisi Islemleri";
475:        this.label12.TextAlign = System.Drawing.ContentAlignment.TopCenter;
476:        //
477:        // btnEPostaDogrula
478:        //
479:        this.btnEPostaDogrula.Location = new System.Drawing.Point(297, 52);
480:        this.btnEPostaDogrula.Margin = new System.Windows.Forms.Padding(4, 4, 4, 4);
481:        this.btnEPostaDogrula.Name = "btnEPostaDogrula";
482:        this.btnEPostaDogrula.Size = new System.Drawing.Size(127, 28);
483:        this.btnEPostaDogrula.TabIndex = 15;
484:        this.btnEPostaDogrula.Text = "E-Posta Dogrula";
485:        this.btnEPostaDogrula.UseVisualStyleBackColor = true;
486:        this.btnEPostaDogrula.Click += new
System.EventHandler(this.btnEPostaDogrula_Click);
487:        //
488:        // btnIbanDogrula
489:        //
490:        this.btnIbanDogrula.Location = new System.Drawing.Point(175, 52);
491:        this.btnIbanDogrula.Margin = new System.Windows.Forms.Padding(4, 4, 4, 4);
492:        this.btnIbanDogrula.Name = "btnIbanDogrula";
493:        this.btnIbanDogrula.Size = new System.Drawing.Size(115, 28);
494:        this.btnIbanDogrula.TabIndex = 14;
495:        this.btnIbanDogrula.Text = "IBAN Dogrula";
496:        this.btnIbanDogrula.UseVisualStyleBackColor = true;
497:        this.btnIbanDogrula.Click += new System.EventHandler(this.btnIbanDogrula_Click);
498:        //
499:        // label13
500:        //
501:        this.label13.AutoSize = true;
502:        this.label13.Location = new System.Drawing.Point(24, 96);
503:        this.label13.Margin = new System.Windows.Forms.Padding(4, 0, 4, 0);
504:        this.label13.Name = "label13";
505:        this.label13.Size = new System.Drawing.Size(48, 17);
506:        this.label13.TabIndex = 13;
507:        this.label13.Text = "Sonuç";
508:        //
509:        // lbKisiSonuc
510:        //
511:        this.lbKisiSonuc.FormattingEnabled = true;
512:        this.lbKisiSonuc.HorizontalScrollbar = true;
513:        this.lbKisiSonuc.ItemHeight = 16;
514:        this.lbKisiSonuc.Location = new System.Drawing.Point(83, 96);
515:        this.lbKisiSonuc.Margin = new System.Windows.Forms.Padding(4, 4, 4, 4);
516:        this.lbKisiSonuc.Name = "lbKisiSonuc";
517:        this.lbKisiSonuc.Size = new System.Drawing.Size(337, 100);
518:        this.lbKisiSonuc.TabIndex = 12;
519:        //
520:        // btnTcKimlikDogrula
521:        //
522:        this.btnTcKimlikDogrula.Location = new System.Drawing.Point(28, 52);
523:        this.btnTcKimlikDogrula.Margin = new System.Windows.Forms.Padding(4, 4, 4, 4);
524:        this.btnTcKimlikDogrula.Name = "btnTcKimlikDogrula";
525:        this.btnTcKimlikDogrula.Size = new System.Drawing.Size(139, 28);
526:        this.btnTcKimlikDogrula.TabIndex = 11;
```

```
527:        this.btnTcKimlikDogrula.Text = "TC Kimlik Dogrula";
528:        this.btnTcKimlikDogrula.UseVisualStyleBackColor = true;
529:        this.btnTcKimlikDogrula.Click += new
System.EventHandler(this.btnTcKimlikDogrula_Click);
530:        //
531:        // txtKisiMetni
532:        //
533:        this.txtKisiMetni.Location = new System.Drawing.Point(157, 20);
534:        this.txtKisiMetni.Margin = new System.Windows.Forms.Padding(4, 4, 4, 4);
535:        this.txtKisiMetni.Name = "txtKisiMetni";
536:        this.txtKisiMetni.Size = new System.Drawing.Size(264, 22);
537:        this.txtKisiMetni.TabIndex = 4;
538:        //
539:        // label14
540:        //
541:        this.label14.AutoSize = true;
542:        this.label14.Location = new System.Drawing.Point(23, 23);
543:        this.label14.Margin = new System.Windows.Forms.Padding(4, 0, 4, 0);
544:        this.label14.Name = "label14";
545:        this.label14.Size = new System.Drawing.Size(42, 17);
546:        this.label14.TabIndex = 3;
547:        this.label14.Text = "Metin";
548:        //
549:        // pnlMetinIslemleri
550:        //
551:        this.pnlMetinIslemleri.BorderStyle = System.Windows.Forms.BorderStyle.FixedSingle;
552:        this.pnlMetinIslemleri.Controls.Add(this.btnEPostaKontrol);
553:        this.pnlMetinIslemleri.Controls.Add(this.btnTelefonNoDuzenle);
554:        this.pnlMetinIslemleri.Controls.Add(this.btnMetinSifrele);
555:        this.pnlMetinIslemleri.Controls.Add(this.label15);
556:        this.pnlMetinIslemleri.Controls.Add(this.btnTurkceKarakterDuzelt);
557:        this.pnlMetinIslemleri.Controls.Add(this.btnIlkHarfiBuyukYap);
558:        this.pnlMetinIslemleri.Controls.Add(this.label16);
559:        this.pnlMetinIslemleri.Controls.Add(this.lbMetinSonuc);
560:        this.pnlMetinIslemleri.Controls.Add(this.btnIlkHarfleriBuyukYap);
561:        this.pnlMetinIslemleri.Controls.Add(this.txtMetin);
562:        this.pnlMetinIslemleri.Controls.Add(this.label17);
563:        this.pnlMetinIslemleri.Location = new System.Drawing.Point(20, 564);
564:        this.pnlMetinIslemleri.Margin = new System.Windows.Forms.Padding(4, 4, 4, 4);
565:        this.pnlMetinIslemleri.Name = "pnlMetinIslemleri";
566:        this.pnlMetinIslemleri.Size = new System.Drawing.Size(1030, 222);
567:        this.pnlMetinIslemleri.TabIndex = 21;
568:        //
569:        // btnTelefonNoDuzenle
570:        //
571:        this.btnTelefonNoDuzenle.Location = new System.Drawing.Point(671, 52);
572:        this.btnTelefonNoDuzenle.Margin = new System.Windows.Forms.Padding(4, 4, 4, 4);
573:        this.btnTelefonNoDuzenle.Name = "btnTelefonNoDuzenle";
574:        this.btnTelefonNoDuzenle.Size = new System.Drawing.Size(179, 28);
575:        this.btnTelefonNoDuzenle.TabIndex = 20;
576:        this.btnTelefonNoDuzenle.Text = "Telefon No Düzenle";
577:        this.btnTelefonNoDuzenle.UseVisualStyleBackColor = true;
578:        this.btnTelefonNoDuzenle.Click += new
System.EventHandler(this.btnTelefonNoDuzenle_Click);
579:        //
580:        // btnMetinSifrele
581:        //
582:        this.btnMetinSifrele.Location = new System.Drawing.Point(484, 52);
583:        this.btnMetinSifrele.Margin = new System.Windows.Forms.Padding(4, 4, 4, 4);
584:        this.btnMetinSifrele.Name = "btnMetinSifrele";
585:        this.btnMetinSifrele.Size = new System.Drawing.Size(179, 28);
586:        this.btnMetinSifrele.TabIndex = 19;
587:        this.btnMetinSifrele.Text = "Metin Sifrele";
588:        this.btnMetinSifrele.UseVisualStyleBackColor = true;
589:        this.btnMetinSifrele.Click += new System.EventHandler(this.btnMetinSifrele_Click);
590:        //
591:        // label15
592:        //
593:        this.label15.AutoSize = true;
```

```
594:          this.label15.Dock = System.Windows.Forms.DockStyle.Top;
595:          this.label15.Location = new System.Drawing.Point(0, 0);
596:          this.label15.Margin = new System.Windows.Forms.Padding(4, 0, 4, 0);
597:          this.label15.Name = "label15";
598:          this.label15.Size = new System.Drawing.Size(97, 17);
599:          this.label15.TabIndex = 18;
600:          this.label15.Text = "Metin Islemleri";
601:          this.label15.TextAlign = System.Drawing.ContentAlignment.TopCenter;
602:          //
603:          // btnTurkceKarakterDuzelt
604:          //
605:          this.btnTurkceKarakterDuzelt.Location = new System.Drawing.Point(297, 52);
606:          this.btnTurkceKarakterDuzelt.Margin = new System.Windows.Forms.Padding(4, 4, 4, 4);
607:          this.btnTurkceKarakterDuzelt.Name = "btnTurkceKarakterDuzelt";
608:          this.btnTurkceKarakterDuzelt.Size = new System.Drawing.Size(179, 28);
609:          this.btnTurkceKarakterDuzelt.TabIndex = 15;
610:          this.btnTurkceKarakterDuzelt.Text = "Türkçe Karakter Düzelt";
611:          this.btnTurkceKarakterDuzelt.UseVisualStyleBackColor = true;
612:          this.btnTurkceKarakterDuzelt.Click += new
System.EventHandler(this.btnTurkceKarakterDuzelt_Click);
613:          //
614:          // btnIlkHarfiBuyukYap
615:          //
616:          this.btnIlkHarfiBuyukYap.Location = new System.Drawing.Point(175, 52);
617:          this.btnIlkHarfiBuyukYap.Margin = new System.Windows.Forms.Padding(4, 4, 4, 4);
618:          this.btnIlkHarfiBuyukYap.Name = "btnIlkHarfiBuyukYap";
619:          this.btnIlkHarfiBuyukYap.Size = new System.Drawing.Size(115, 28);
620:          this.btnIlkHarfiBuyukYap.TabIndex = 14;
621:          this.btnIlkHarfiBuyukYap.Text = "Ilk Harf Büyük";
622:          this.btnIlkHarfiBuyukYap.UseVisualStyleBackColor = true;
623:          this.btnIlkHarfiBuyukYap.Click += new
System.EventHandler(this.btnIlkHarfiBuyukYap_Click);
624:          //
625:          // label16
626:          //
627:          this.label16.AutoSize = true;
628:          this.label16.Location = new System.Drawing.Point(24, 96);
629:          this.label16.Margin = new System.Windows.Forms.Padding(4, 0, 4, 0);
630:          this.label16.Name = "label16";
631:          this.label16.Size = new System.Drawing.Size(48, 17);
632:          this.label16.TabIndex = 13;
633:          this.label16.Text = "Sonuç";
634:          //
635:          // lbMetinSonuc
636:          //
637:          this.lbMetinSonuc.FormattingEnabled = true;
638:          this.lbMetinSonuc.HorizontalScrollbar = true;
639:          this.lbMetinSonuc.ItemHeight = 16;
640:          this.lbMetinSonuc.Location = new System.Drawing.Point(83, 96);
641:          this.lbMetinSonuc.Margin = new System.Windows.Forms.Padding(4, 4, 4, 4);
642:          this.lbMetinSonuc.Name = "lbMetinSonuc";
643:          this.lbMetinSonuc.Size = new System.Drawing.Size(868, 100);
644:          this.lbMetinSonuc.TabIndex = 12;
645:          //
646:          // btnIlkHarfleriBuyukYap
647:          //
648:          this.btnIlkHarfleriBuyukYap.Location = new System.Drawing.Point(28, 52);
649:          this.btnIlkHarfleriBuyukYap.Margin = new System.Windows.Forms.Padding(4, 4, 4, 4);
650:          this.btnIlkHarfleriBuyukYap.Name = "btnIlkHarfleriBuyukYap";
651:          this.btnIlkHarfleriBuyukYap.Size = new System.Drawing.Size(139, 28);
652:          this.btnIlkHarfleriBuyukYap.TabIndex = 11;
653:          this.btnIlkHarfleriBuyukYap.Text = "Ilk Harfler Büyük";
654:          this.btnIlkHarfleriBuyukYap.UseVisualStyleBackColor = true;
655:          this.btnIlkHarfleriBuyukYap.Click += new
System.EventHandler(this.btnIlkHarfleriBuyukYap_Click);
656:          //
657:          // txtMetin
658:          //
659:          this.txtMetin.Location = new System.Drawing.Point(159, 15);
```

```
660:          this.txtMetin.Margin = new System.Windows.Forms.Padding(4, 4, 4, 4);
661:          this.txtMetin.Name = "txtMetin";
662:          this.txtMetin.Size = new System.Drawing.Size(792, 22);
663:          this.txtMetin.TabIndex = 4;
664:          //
665:          // label17
666:          //
667:          this.label17.AutoSize = true;
668:          this.label17.Location = new System.Drawing.Point(23, 23);
669:          this.label17.Margin = new System.Windows.Forms.Padding(4, 0, 4, 0);
670:          this.label17.Name = "label17";
671:          this.label17.Size = new System.Drawing.Size(42, 17);
672:          this.label17.TabIndex = 3;
673:          this.label17.Text = "Metin";
674:          //
675:          // btnEPostaKontrol
676:          //
677:          this.btnEPostaKontrol.Location = new System.Drawing.Point(858, 52);
678:          this.btnEPostaKontrol.Margin = new System.Windows.Forms.Padding(4);
679:          this.btnEPostaKontrol.Name = "btnEPostaKontrol";
680:          this.btnEPostaKontrol.Size = new System.Drawing.Size(160, 28);
681:          this.btnEPostaKontrol.TabIndex = 21;
682:          this.btnEPostaKontrol.Text = "E-Posta Geçerli Mi?";
683:          this.btnEPostaKontrol.UseVisualStyleBackColor = true;
684:          this.btnEPostaKontrol.Click += new
System.EventHandler(this.btnEPostaKontrol_Click);
685:          //
686:          // FrmAnaSayfa
687:          //
688:          this.AutoScaleDimensions = new System.Drawing.SizeF(8F, 16F);
689:          this.AutoScaleMode = System.Windows.Forms.AutoScaleMode.Font;
690:          this.AutoSizeMode = System.Windows.Forms.AutoSizeMode.GrowAndShrink;
691:          this.ClientSize = new System.Drawing.Size(1067, 844);
692:          this.Controls.Add(this.pnlMetinIslemleri);
693:          this.Controls.Add(this.pnlKisiIslemleri);
694:          this.Controls.Add(this.pnlSayiIslemleri);
695:          this.Controls.Add(this.pnlSifreIslemleri);
696:          this.Controls.Add(this.pnlTarihIslemleri);
697:          this.Controls.Add(this.label1);
698:          this.Controls.Add(this.cmbIslemTuru);
699:          this.Margin = new System.Windows.Forms.Padding(4, 4, 4, 4);
700:          this.Name = "FrmAnaSayfa";
701:          this.StartPosition = System.Windows.Forms.FormStartPosition.CenterScreen;
702:          this.Text = "Islemler Ekrani";
703:          this.Load += new System.EventHandler(this.FrmAnaSayfa_Load);
704:          this.pnlTarihIslemleri.ResumeLayout(false);
705:          this.pnlTarihIslemleri.PerformLayout();
706:          this.pnlSifreIslemleri.ResumeLayout(false);
707:          this.pnlSifreIslemleri.PerformLayout();
708:          this.pnlSayiIslemleri.ResumeLayout(false);
709:          this.pnlSayiIslemleri.PerformLayout();
710:          this.pnlKisiIslemleri.ResumeLayout(false);
711:          this.pnlKisiIslemleri.PerformLayout();
712:          this.pnlMetinIslemleri.ResumeLayout(false);
713:          this.pnlMetinIslemleri.PerformLayout();
714:          this.ResumeLayout(false);
715:          this.PerformLayout();
716: }
```

1.1.4.1.2 Program Class

Class Hierarchy

CSharpHelperMethods.UI.Program

C#

```
internal static class Program;
```

File

Program.cs (⧉ see page 112)

Description

This is class CSharpHelperMethods.UI.Program.

Program Methods

	Name	Description
⇥•▪§	Main (⧉ see page 77)	The main entry point for the application.

1.1.4.1.2.1 Program Methods

1.1.4.1.2.1.1 Program.Main Method

The main entry point for the application.

C#

```
[STAThread]
private static void Main();
```

Body Source

```
1: [STAThread]
2: private static void Main()
3: {
4:     Application.EnableVisualStyles();
5:     Application.SetCompatibleTextRenderingDefault(false);
6:     Application.Run(new FrmAnaSayfa());
7: }
```

1.2 Files

The following table lists files in this documentation.

Files

Name	Description
AssemblyInfo.cs (⧉ see page 78)	This is file AssemblyInfo.cs.
CSharpHelperMethods.Library.csproj (⧉ see page 79)	This is file CSharpHelperMethods.Library.csproj.
CSharpHelperMethods.sln (⧉ see page 79)	This is file CSharpHelperMethods.sln.
CSharpHelperMethods.Test.csproj (⧉ see page 79)	This is file CSharpHelperMethods.Test.csproj.
CSharpHelperMethods.UI.csproj (⧉ see page 79)	This is file CSharpHelperMethods.UI.csproj.
DataTableIslemleri.cs (⧉ see page 79)	This is file DataTableIslemleri.cs.
DosyaIslemleri.cs (⧉ see page 81)	This is file DosyaIslemleri.cs.
FrmAnaSayfa.cs (⧉ see page 83)	This is file FrmAnaSayfa.cs.
FrmAnaSayfa.Designer.cs (⧉ see page 93)	This is file FrmAnaSayfa.Designer.cs.
KisiIslemleri.cs (⧉ see page 105)	This is file KisiIslemleri.cs.
MetinIslemleri.cs (⧉ see page 107)	This is file MetinIslemleri.cs.
MetinIslemleriTest.cs (⧉ see page 111)	This is file MetinIslemleriTest.cs.
Program.cs (⧉ see page 112)	This is file Program.cs.

1.2.1 AssemblyInfo.cs

This is file AssemblyInfo.cs.

Body Source

```
 1: ?using System.Reflection;
 2: using System.Runtime.InteropServices;
 3:
 4: // General Information about an assembly is controlled through the following
 5: // set of attributes. Change these attribute values to modify the information
 6: // associated with an assembly.
 7: [assembly: AssemblyTitle("CSharpHelperMethods.UI")]
 8: [assembly: AssemblyDescription("")]
 9: [assembly: AssemblyConfiguration("")]
10: [assembly: AssemblyCompany("")]
11: [assembly: AssemblyProduct("CSharpHelperMethods.UI")]
12: [assembly: AssemblyCopyright("Copyright ©  2021")]
13: [assembly: AssemblyTrademark("")]
14: [assembly: AssemblyCulture("")]
15:
16: // Setting ComVisible to false makes the types in this assembly not visible
17: // to COM components.  If you need to access a type in this assembly from
18: // COM, set the ComVisible attribute to true on that type.
19: [assembly: ComVisible(false)]
20:
21: // The following GUID is for the ID of the typelib if this project is exposed to COM
22: [assembly: Guid("be02934f-770b-411c-aa8c-43b2efa18625")]
23:
24: // Version information for an assembly consists of the following four values:
25: //
26: //      Major Version
27: //      Minor Version
28: //      Build Number
29: //      Revision
30: //
31: // You can specify all the values or you can default the Build and Revision Numbers
32: // by using the '*' as shown below:
33: // [assembly: AssemblyVersion("1.0.*")]
34: [assembly: AssemblyVersion("1.0.0.0")]
35: [assembly: AssemblyFileVersion("1.0.0.0")]
```

1.2.2 CSharpHelperMethods.Library.csproj

This is file CSharpHelperMethods.Library.csproj.

1.2.3 CSharpHelperMethods.sln

This is file CSharpHelperMethods.sln.

1.2.4 CSharpHelperMethods.Test.csproj

This is file CSharpHelperMethods.Test.csproj.

1.2.5 CSharpHelperMethods.UI.csproj

This is file CSharpHelperMethods.UI.csproj.

1.2.6 DataTableIslemleri.cs

This is file DataTableIslemleri.cs.

Body Source

```
 1: ?using System;
 2: using System.Collections.Generic;
 3: using System.Data;
 4: using System.Reflection;
 5:
 6: namespace CSharpHelperMethods.Library
 7: {
 8:     /// <summary>
 9:     /// Raporlama için lazim olan DataTable vb. verilerin düzenlenmesini saglayan
siniflardir.
10:     /// </summary>
11:     public static class DataTableIslemleri
12:     {
13:         /// <summary>
14:         /// Parametre olarak girilen herhangi bir sinif türünden listeyi
15:         /// Idael bir sekilde Nullable alan içermeyen bir DataTable'a çevirmek için
kullanilan metottur.
16:         /// </summary>
17:         /// <typeparam name="T">Generic Nesne Tipi</typeparam>
18:         /// <param name="liste">Liste Bilgisi</param>
19:         /// <returns>DataTable cinsinde Nesne</returns>
20:         public static DataTable ToDataTable<T>(List<T> liste)
21:         {
22:             try
23:             {
```

```
 24:                          var sonuc = new DataTable(typeof(T).Name);
 25:
 26:                          var ozellikler = typeof(T).GetFields(BindingFlags.Public |
BindingFlags.CreateInstance | BindingFlags.Default | BindingFlags.DeclaredOnly |
BindingFlags.ExactBinding | BindingFlags.FlattenHierarchy | BindingFlags.IgnoreReturn |
BindingFlags.Instance | BindingFlags.InvokeMethod | BindingFlags.NonPublic |
BindingFlags.OptionalParamBinding | BindingFlags.SuppressChangeType);
 27:
 28:                          var ozellikler2 = typeof(T).GetProperties(BindingFlags.Public |
BindingFlags.CreateInstance | BindingFlags.Default | BindingFlags.DeclaredOnly |
BindingFlags.ExactBinding | BindingFlags.FlattenHierarchy | BindingFlags.IgnoreReturn |
BindingFlags.Instance | BindingFlags.InvokeMethod | BindingFlags.NonPublic |
BindingFlags.OptionalParamBinding | BindingFlags.SuppressChangeType);
 29:
 30:                          foreach (var ozellik in ozellikler)
 31:                          {
 32:                              var varMi = false;
 33:                              foreach (var ozellik2 in ozellikler2)
 34:                              {
 35:                                  if (!ozellik.Name.Contains(ozellik2.Name)) continue;
 36:                                  varMi = true;
 37:                                  var ozellikAdi = ozellik.Name.Replace("<", "");
 38:                                  var ozellikDizi = ozellikAdi.Split('>');
 39:                                  var ozellikTemp = ozellikDizi[0].Trim();
 40:                                  //ozellikTemp = ozellikTemp.Substring(0, ozellikTemp.Length -
1);
 41:                                  if (ozellikTemp == ozellik2.Name)
 42:                                  {
 43:                                      if (sonuc.Columns.IndexOf(new DataColumn(ozellik2.Name))
>= 0) continue;
 44:                                      sonuc.Columns.Add(ozellik2.Name,
BaseType(ozellik2.PropertyType));
 45:                                      break;
 46:                                  }
 47:
 48:                                  if (sonuc.Columns.IndexOf(new DataColumn(ozellikTemp)) >= 0)
continue;
 49:                                  sonuc.Columns.Add(ozellikTemp,
BaseType(ozellik2.PropertyType));
 50:                                  break;
 51:                              }
 52:
 53:                              if (varMi) continue;
 54:                              if (sonuc.Columns.IndexOf(new DataColumn(ozellik.Name)) < 0)
 55:                              {
 56:                                  sonuc.Columns.Add(ozellik.Name, BaseType(ozellik.FieldType));
 57:                              }
 58:
 59:                          }
 60:
 61:                          foreach (var nesne in liste)
 62:                          {
 63:                              var degerler = new object[ozellikler.Length];
 64:
 65:                              for (var i = 0; i < ozellikler.Length; i++)
 66:                              {
 67:                                  var varMi = false;
 68:                                  foreach (var item in ozellikler2)
 69:                                  {
 70:                                      if (!ozellikler[i].Name.Contains(item.Name)) continue;
 71:                                      varMi = true;
 72:                                      degerler[i] = item.GetValue(nesne, null);
 73:                                      break;
 74:                                  }
 75:                                  if (!varMi)
 76:                                  {
 77:                                      degerler[i] = ozellikler[i].GetValue(nesne);
 78:                                  }
 79:
```

```
80:                          }
81:
82:                          sonuc.Rows.Add(degerler);
83:                      }
84:
85:                      return sonuc;
86:                  }
87:              catch (Exception)
88:              {
89:                  return null;
90:              }
91:          }
92:
93:          /// <summary>
94:          /// Nesnenin içindeki her hangi bir degerin (DateTime, decimal vb.)
95:          /// Eger Nullable ise, Nullable olmamasini saglayan metottur
96:          /// (Çünkü Crystal Report, Telerik vb. Nullable alan kabul etmiyor)
97:          /// </summary>
98:          /// <param name="oType">Veri Tipi</param>
99:          /// <returns>Verinin Ideal Tipi</returns>
100:         private static Type BaseType(Type oType) => oType != null && oType.IsValueType
&& oType.IsGenericType && oType.GetGenericTypeDefinition() == typeof(Nullable<>) ?
Nullable.GetUnderlyingType(oType) : oType;
101:     }
102: }
```

Namespaces

Name	Description
CSharpHelperMethods (see page 1)	This is namespace CSharpHelperMethods.

1.2.7 DosyaIslemleri.cs

This is file DosyaIslemleri.cs.

Body Source

```
1: ?using System;
2: using System.IO;
3:
4: namespace CSharpHelperMethods.Library
5: {
6:     /// <summary>
7:     /// Dosya islemleri ile ilgili yardimici sinif
8:     /// </summary>
9:     public static class DosyaIslemleri
10:    {
11:        /// <summary>
12:        /// Dosya uzantisindan dosyanin türünün ne oldugunu getiren metottur
13:        /// </summary>
14:        /// <param name="extension">Dosya Uzantisi</param>
15:        /// <returns>Dosya Türü Bilgisi</returns>
16:        public static string GetirIcerikTipiDosyaUzantisiIle(string extension)
17:        {
18:            var contentType = String.Empty;
19:            extension = extension.Replace(".", "").ToLower();
20:            switch (extension)
21:            {
22:                case "pdf": contentType = "application/pdf"; break;
23:                case "jpeg": contentType = "image/jpeg"; break;
24:                case "jpg": contentType = "image/jpg"; break;
25:                case "png": contentType = "image/png"; break;
26:                case "bmp": contentType = "image/bmp"; break;
27:                case "doc": contentType = "application/msword"; break;
28:                case "pptx": contentType =
```

```
          "application/vnd.openxmlformats-officedocument.presentationml.presentation"; break;
 29:                 case "xls": contentType = "application/vnd.ms-excel"; break;
 30:                 case "txt": contentType = "text/plain"; break;
 31:                 case "xlsx": contentType =
          "application/vnd.openxmlformats-officedocument.spreadsheetml.sheet"; break;
 32:                 case "rtf": contentType = "application/rtf"; break;
 33:                 case "docx": contentType =
          "application/vnd.openxmlformats-officedocument.wordprocessingml.document"; break;
 34:                 case "ppt": contentType = "application/vnd.ms-powerpoint"; break;
 35:                 case "zip": contentType = "application/zip"; break;
 36:                 case "rar": contentType = "application/x-rar-compressed"; break;
 37:                 case "7z": contentType = "application/x-7z-compressed"; break;
 38:                 default: contentType = "application/octet-stream"; break;
 39:             }
 40:
 41:             return contentType;
 42:         }
 43:
 44:         /// <summary>
 45:         /// Dosyanin türünden dosyanin uzantisinin ne oldugunu getiren metottur
 46:         /// </summary>
 47:         /// <param name="mimeType">Mime Type</param>
 48:         /// <returns>Dosya Türü Bilgisi</returns>
 49:         public static string GetirDosyaUzantisiMimeTypeIle(string mimeType)
 50:         {
 51:             string dosyaUzantisi;
 52:             mimeType = mimeType.ToLower();
 53:             switch (mimeType)
 54:             {
 55:                 case "application/pdf": dosyaUzantisi = "pdf"; break;
 56:                 case "image/jpeg": dosyaUzantisi = "jpeg"; break;
 57:                 case "image/jpg": dosyaUzantisi = "jpg"; break;
 58:                 case "image/png": dosyaUzantisi = "png"; break;
 59:                 case "image/bmp": dosyaUzantisi = "bmp"; break;
 60:                 case "application/msword": dosyaUzantisi = "doc"; break;
 61:                 case "application / vnd.openxmlformats -
          officedocument.presentationml.presentation": dosyaUzantisi = "pptx"; break;
 62:                 case "application/vnd.ms-excel": dosyaUzantisi = "xls"; break;
 63:                 case "text/plain": dosyaUzantisi = "txt"; break;
 64:                 case
          "application/vnd.openxmlformats-officedocument.spreadsheetml.sheet": dosyaUzantisi =
          "xlsx"; break;
 65:                 case
          "application/vnd.openxmlformats-officedocument.wordprocessingml.document": dosyaUzantisi =
          "docx"; break;
 66:                 case "text/xml": dosyaUzantisi = "xml"; break;
 67:                 case "application/vnd.ms-powerpoint": dosyaUzantisi = "ppt"; break;
 68:                 case "application/zip": dosyaUzantisi = "zip"; break;
 69:                 case "application/x-rar-compressed": dosyaUzantisi = "rar"; break;
 70:                 case "application/x-7z-compressed": dosyaUzantisi = "7z"; break;
 71:                 default: dosyaUzantisi = "error"; break;
 72:             }
 73:
 74:             return dosyaUzantisi;
 75:         }
 76:
 77:         /// <summary>
 78:         /// Sisteme yüklenecek dosyanin belirlenen kriterlere uyup uymadigini kontrol
     eden metottur.
 79:         /// Dosyanin uzantisi istedigimiz formatta degilse yanlis olarak geri
     döndürecektir.s
 80:         /// </summary>
 81:         /// <param name="extension">Dosya Uzantisi</param>
 82:         /// <returns>Eklenecek Dosyanin Uygun Olup Olmadigi Bilgisi</returns>
 83:         public static bool DosyaUzantisiUygunMu(string extension)
 84:         {
 85:             extension = extension.ToLower();
 86:
 87:             return extension == ".jpg" || extension == ".bmp" || extension == ".png"
```

```
     || extension == ".doc" || extension == ".docx" || extension == ".ppt" || extension ==
     ".pptx" || extension == ".xls" || extension == ".xlsx" || extension == ".rar" || extension
     == ".zip" || extension == ".7z" || extension == ".pdf" || extension == ".txt" || extension
     == ".rtf";
 88:          }
 89:
 90:          /// <summary>
 91:          /// Dosya sistemindeki bir dosyanin baska bir program tarafindan o anda
     kullanimda olup olmadiginin kontol edildigi bir metottur.
 92:          /// </summary>
 93:          /// <param name="dosya">Dosya Bilgisi</param>
 94:          /// <returns>Evet veya Hayir Bilgisi</returns>
 95:          public static bool DosyaKilitliMi(FileInfo dosya)
 96:          {
 97:              FileStream stream = null;
 98:              try
 99:              {
100:                  stream = dosya.Open(FileMode.Open, FileAccess.ReadWrite,
     FileShare.None);
101:              }
102:              catch (IOException)
103:              {
104:                  // eger dosya baska bir uygulama tarafindan kullaniliyorsa
105:                  // burada true dönüyor ki
106:                  // kullanici bu metottan gelen bilgiye göre davransin
107:                  return true;
108:              }
109:              finally
110:              {
111:                  stream?.Close();
112:              }
113:
114:              //file is not locked
115:              return false;
116:          }
117:
118:          /// <summary>
119:          /// Bu metot ile gönderdiginiz Dizin yolunda verdiginiz isimde bir klasör yoksa
120:          /// Otomatik olarak olusturmayi sagliyor
121:          /// </summary>
122:          /// <param name="dizinYolu">Dizin Yolu Bilgisi</param>
123:          /// <returns>Evet veya Hayir Bilgisi</returns>
124:          public static bool GuvenliDizinOlustur(string dizinYolu)
125:          {
126:              var isExists = Directory.Exists(dizinYolu);
127:
128:              if (!isExists)
129:                  Directory.CreateDirectory(dizinYolu);
130:              return isExists;
131:          }
132:      }
133: }
```

Namespaces

Name	Description
CSharpHelperMethods (see page 1)	This is namespace CSharpHelperMethods.

1.2.8 FrmAnaSayfa.cs

This is file FrmAnaSayfa.cs.

Body Source

```
  1: ?using System;
```

```
 2: using System.Collections.Generic;
 3: using System.Drawing;
 4: using System.Linq;
 5: using CSharpHelperMethods.Library;
 6: using System.Text;
 7: using System.Windows.Forms;
 8:
 9: namespace CSharpHelperMethods.UI
10: {
11:     public partial class FrmAnaSayfa : Form
12:     {
13:         public FrmAnaSayfa()
14:         {
15:             InitializeComponent();
16:         }
17:
18:         #region Formun Yüklenmesi, Panel Seçimi ve ListBox Düzenlemeleri
19:         /// <summary>
20:         /// Form Ilk Yüklendigi zaman ekrandaki bütün panellerin erisilemez hale
getirilmesi
21:         /// saglaniyor Bu metotta, çünkü hangi panelin aktiflesecegininin
22:         /// Combobox üzerinden seçilmesi lazim
23:         /// </summary>
24:         /// <param name="sender">The sender info (For example Main Form)</param>
25:         /// <param name="e">Event Arguments</param>
26:         private void FrmAnaSayfa_Load(object sender, EventArgs e)
27:         {
28:             cmbIslemTuru.DropDownStyle = ComboBoxStyle.DropDownList;
29:             PaneliErisilebilirKil("Hicbiri");
30:             ListBoxlariCokluSatirliYap();
31:         }
32:
33:         /// <summary>
34:         /// Islem Türü Combobox'inda seçilen degere göre
35:         /// Ekrandaki panellerden ilgili olan haricinde hepsinin erisilemez hale
getirilmesi
36:         /// Eger bir sey seçilmemisse, tamaminin erisilemez hale getirilmesi için
37:         /// Gerekli islemlerin gerçeklestirildigi metottur.
38:         /// </summary>
39:         /// <param name="sender">The sender info (For example Main Form)</param>
40:         /// <param name="e">Event Arguments</param>
41:         private void cmbIslemTuru_SelectedIndexChanged(object sender, EventArgs e)
42:         {
43:             #region Combobox'tan Seçilen Degere Göre Ilgili Paneli Aktif Hale Getirme
44:             switch (cmbIslemTuru.SelectedIndex)
45:             {
46:                 // Metin Islemleri
47:                 case 0:
48:                     {
49:                         PaneliErisilebilirKil("pnlMetinIslemleri");
50:                         break;
51:                     }
52:                 // Tarih Islemleri
53:                 case 1:
54:                     {
55:                         PaneliErisilebilirKil("pnlTarihIslemleri");
56:                         break;
57:                     }
58:                 // Sayi Islemleri
59:                 case 2:
60:                     {
61:                         PaneliErisilebilirKil("pnlSayiIslemleri");
62:                         break;
63:                     }
64:                 // Sifre Islemleri
65:                 case 3:
66:                     {
67:                         PaneliErisilebilirKil("pnlSifreIslemleri");
68:                         break;
```

```
 69:                            }
 70:                       // Kisi Islemleri
 71:                       case 4:
 72:                            {
 73:                                PaneliErisilebilirKil("pnlKisiIslemleri");
 74:                                break;
 75:                            }
 76:                       // Bunlar haricinde herhangi bir sey
 77:                       default:
 78:                            {
 79:                                PaneliErisilebilirKil("Hicbiri");
 80:                                break;
 81:                            }
 82:                   }
 83:               #endregion
 84:          }
 85:
 86:          /// <summary>
 87:          /// Parametre olarak verilen panelin aktif hale getirilmesi
 88:          /// Diger panellerin deaktif edilmesi için hazirlanmis metottur
 89:          /// </summary>
 90:          /// <param name="panelAdi">Panel Adi Bilgisi</param>
 91:          private void PaneliErisilebilirKil(string panelAdi)
 92:          {
 93:               foreach (Control formElement in Controls)
 94:               {
 95:                   if (formElement.Name == panelAdi)
 96:                       formElement.Enabled = true;
 97:                   else if (formElement.Name.StartsWith("pnl"))
 98:                       formElement.Enabled = false;
 99:               }
100:          }
101:
102:          /// <summary>
103:          /// Tüm Panelleri gezerek panellerin içerisindeki listboxlarin
104:          /// Multiline özelligine kavusmasi için
105:          /// Gerekli islemlerin düzenlendigi metottur.
106:          /// </summary>
107:          private void ListBoxlariCokluSatirliYap()
108:          {
109:               foreach (Control formElement in Controls)
110:               {
111:                   if (formElement.GetType() != typeof(Panel)) continue;
112:                   var listBoxList = formElement.Controls.OfType<ListBox>().ToList();
113:                   foreach (var listBox in listBoxList)
114:                   {
115:                       listBox.DrawMode = DrawMode.OwnerDrawVariable;
116:                       listBox.MeasureItem += listBox_MeasureItem;
117:                       listBox.DrawItem += listBox_DrawItem;
118:                   }
119:               }
120:          }
121:
122:          /// <summary>
123:          /// This method is used to meaure the length of the item in listbox and make
it multi line
124:          /// without using scrollbar and let the user see more efficiently
125:          /// </summary>
126:          /// <param name="sender">The sender info (For example Main Form)</param>
127:          /// <param name="e">Event Arguments</param>
128:          private static void listBox_MeasureItem(object sender, MeasureItemEventArgs e)
129:          {
130:               if (!(sender is ListBox listBox)) return;
131:               e.ItemHeight = (int) e.Graphics
132:                    .MeasureString(listBox.Items[e.Index].ToString(), listBox.Font,
listBox.Width).Height;
133:          }
134:
135:          /// <summary>
```

```
136:          /// This method is used to meaure the length of the item in listbox and make
it multi line
137:          /// without using scrollbar and let the user see more efficiently
138:          /// </summary>
139:          /// <param name="sender">The sender info (For example Main Form)</param>
140:          /// <param name="e">Event Arguments</param>
141:          private static void listBox_DrawItem(object sender, DrawItemEventArgs e)
142:          {
143:              if (!(sender is ListBox listBox)) return;
144:              e.DrawBackground();
145:              e.DrawFocusRectangle();
146:              e.Graphics.DrawString(listBox.Items[e.Index].ToString(), e.Font, new
SolidBrush(e.ForeColor), e.Bounds);
147:          }
148:
149:          #endregion
150:
151:          #region Tarih Islemleri
152:
153:          /// <summary>
154:          /// Paneldeki Baslangiç Tarihi seçili ve tarih girilmisse,
155:          /// o tarihten bugüne kaç yas oldugunun hesaplandigi metottur.
156:          /// Ilgili butona tiklandigi zaman çalisir
157:          /// </summary>
158:          /// <param name="sender">The sender info (For example Main Form)</param>
159:          /// <param name="e">Event Arguments</param>
160:          private void btnYasHesapla_Click(object sender, EventArgs e)
161:          {
162:              if (!dtpBaslangicTarihi.Checked)
163:                  MessageBox.Show(Sabitler.BaslangicTarihiGirilmedi);
164:              else
165:              {
166:                  lbSonuc.Items.Clear();
167:                  var sb = new StringBuilder();
168:                  sb.Append("Dogum tarihi seçilen kisi ");
169:                  sb.Append(TarihIslemleri.YasHesapla(dtpBaslangicTarihi.Value));
170:                  sb.Append(" yasinin içindedir");
171:                  lbSonuc.Items.Add(sb.ToString());
172:              }
173:          }
174:
175:          /// <summary>
176:          /// Paneldeki Baslangiç Tarihi seçili ve tarih girilmisse,
177:          /// o tarihten bugüne kaç yas oldugunun yil, ay ve gün
178:          /// olarak metinsel bir sekilde hesaplandigi metottur.
179:          /// Ilgili butona tiklandigi zaman çalisir
180:          /// </summary>
181:          /// <param name="sender">The sender info (For example Main Form)</param>
182:          /// <param name="e">Event Arguments</param>
183:          private void btnYasHesaplaMetinsel_Click(object sender, EventArgs e)
184:          {
185:              if (!dtpBaslangicTarihi.Checked)
186:                  MessageBox.Show(Sabitler.BaslangicTarihiGirilmedi);
187:              else
188:              {
189:                  lbSonuc.Items.Clear();
190:                  var sb = new StringBuilder();
191:                  sb.Append(TarihIslemleri.YasHesaplaMetinsel(dtpBaslangicTarihi.Value));
192:                  lbSonuc.Items.Add(sb.ToString());
193:              }
194:          }
195:
196:          /// <summary>
197:          /// Paneldeki Baslangiç Tarihi ve Bitis Tarihi alanlari seçili ve tarih
girilmisse,
198:          /// Söz konusu tarihler arasinda kaç yil, kaç ay ve gün oldugunun bilgisinin
hazirlandigi metottur.
199:          /// Ilgili butona tiklandigi zaman çalisir
200:          /// </summary>
```

```
201:             /// <param name="sender">The sender info (For example Main Form)</param>
202:             /// <param name="e">Event Arguments</param>
203:             private void btnTarihAraligi_Click(object sender, EventArgs e)
204:             {
205:                 if (!dtpBaslangicTarihi.Checked || !dtpBitisTarihi.Checked)
206:                     MessageBox.Show(Sabitler.BaslangicVeyaBitisTarihiGirilmedi);
207:                 else if (dtpBaslangicTarihi.Value.Date > dtpBitisTarihi.Value.Date)
208:                     MessageBox.Show(Sabitler.BaslangicTarihiBuyuk);
209:                 else
210:                 {
211:                     lbSonuc.Items.Clear();
212:                     var sb = new StringBuilder();
213:                     sb.Append("Iki tarih arasi: ");
214:
sb.Append(TarihIslemleri.TarihAraligiHesapla(dtpBaslangicTarihi.Value.Date,
dtpBitisTarihi.Value.Date));
215:                     lbSonuc.Items.Add(sb.ToString());
216:                 }
217:             }
218:
219:             /// <summary>
220:             /// Paneldeki Baslangiç Tarihi ve Bitis Tarihi alanlari seçili ve tarih
girilmisse,
221:             /// Söz konusu tarihler arasinda kaç ay oldugunun bilgisinin hazirlandigi
metottur.
222:             /// Ilgili butona tiklandigi zaman çalisir
223:             /// </summary>
224:             /// <param name="sender">The sender info (For example Main Form)</param>
225:             /// <param name="e">Event Arguments</param>
226:             private void btnAyFarki_Click(object sender, EventArgs e)
227:             {
228:                 if (!dtpBaslangicTarihi.Checked || !dtpBitisTarihi.Checked)
229:                     MessageBox.Show(Sabitler.BaslangicVeyaBitisTarihiGirilmedi);
230:                 else if (dtpBaslangicTarihi.Value.Date > dtpBitisTarihi.Value.Date)
231:                     MessageBox.Show(Sabitler.BaslangicTarihiBuyuk);
232:                 else
233:                 {
234:                     lbSonuc.Items.Clear();
235:                     var sb = new StringBuilder();
236:                     sb.Append("Ay Farki: ");
237:                     sb.Append(TarihIslemleri.AyFarki(dtpBaslangicTarihi.Value.Date,
dtpBitisTarihi.Value.Date));
238:                     lbSonuc.Items.Add(sb.ToString());
239:                 }
240:             }
241:             #endregion
242:
243:             #region Sifre Islemleri
244:             /// <summary>
245:             /// Paneldeki metin kutusu dolu ise
246:             /// Girilen metnin MD5 algoritmasi ile sifrelenmis halini
247:             /// Listbox üzerine yazdiran metottur.
248:             /// </summary>
249:             /// <param name="sender">The sender info (For example Main Form)</param>
250:             /// <param name="e">Event Arguments</param>
251:             private void btnMd5Hash_Click(object sender, EventArgs e)
252:             {
253:                 if (string.IsNullOrEmpty(txtSifrelenecekMetin.Text.Trim()))
254:                     MessageBox.Show(Sabitler.SifrelenecekMetinGirilmedi);
255:                 else
256:                 {
257:                     lbSifreSonuc.Items.Clear();
258:                     var sb = new StringBuilder();
259:                     sb.Append("Girilen Metnin MD5 Karsiligi: ");
260:
sb.Append(SifrelemeIslemleri.HesaplaMD5(txtSifrelenecekMetin.Text.Trim()));
261:                     lbSifreSonuc.Items.Add(sb.ToString());
262:                 }
263:             }
```

```
264:
265:           /// <summary>
266:           /// Paneldeki metin kutusu dolu ise
267:           /// Girilen metnin SHA 256 algoritmasi ile sifrelenmis halini
268:           /// Listbox üzerine yazdiran metottur.
269:           /// </summary>
270:           /// <param name="sender">The sender info (For example Main Form)</param>
271:           /// <param name="e">Event Arguments</param>
272:           private void btnSha256Hash_Click(object sender, EventArgs e)
273:           {
274:               if (string.IsNullOrEmpty(txtSifrelenecekMetin.Text.Trim()))
275:                   MessageBox.Show(Sabitler.SifrelenecekMetinGirilmedi);
276:               else
277:               {
278:                   lbSifreSonuc.Items.Clear();
279:                   var sb = new StringBuilder();
280:                   sb.Append("Girilen Metnin SHA 256 Karsiligi: ");
281:
sb.Append(SifrelemeIslemleri.HesaplaSHA256(txtSifrelenecekMetin.Text.Trim()));
282:                   lbSifreSonuc.Items.Add(sb.ToString());
283:               }
284:           }
285:
286:           /// <summary>
287:           /// Paneldeki metin kutusu dolu ise
288:           /// Girilen metnin SHA 512 algoritmasi ile sifrelenmis halini
289:           /// Listbox üzerine yazdiran metottur.
290:           /// </summary>
291:           /// <param name="sender">The sender info (For example Main Form)</param>
292:           /// <param name="e">Event Arguments</param>
293:           private void btnSha512Hash_Click(object sender, EventArgs e)
294:           {
295:               if (string.IsNullOrEmpty(txtSifrelenecekMetin.Text.Trim()))
296:                   MessageBox.Show(Sabitler.SifrelenecekMetinGirilmedi);
297:               else
298:               {
299:                   lbSifreSonuc.Items.Clear();
300:                   var sb = new StringBuilder();
301:                   sb.Append("Girilen Metnin SHA 512 Karsiligi: ");
302:
sb.Append(SifrelemeIslemleri.HesaplaSHA512(txtSifrelenecekMetin.Text.Trim()));
303:                   lbSifreSonuc.Items.Add(sb.ToString());
304:               }
305:           }
306:
307:           /// <summary>
308:           /// Ilgili butona tiklandiginda rastgele sifre üretilmesini saglayan metottur
309:           /// </summary>
310:           /// <param name="sender">The sender info (For example Main Form)</param>
311:           /// <param name="e">Event Arguments</param>
312:           private void btnRastgeleSifreUret_Click(object sender, EventArgs e)
313:           {
314:               lbSifreSonuc.Items.Clear();
315:               var sb = new StringBuilder();
316:               sb.Append("Rastgele Üretilen Sifre: ");
317:               sb.Append(SifrelemeIslemleri.RastgeleSifreUret());
318:               lbSifreSonuc.Items.Add(sb.ToString());
319:           }
320:
321:           /// <summary>
322:           /// Paneldeki metin kutusu dolu ise
323:           /// Burada yazilan metnin bir sifre olarak belirli kriterlere uygun olup
olmadiginin
324:           /// Gösterilmesini saglayan bir metottur
325:           /// </summary>
326:           /// <param name="sender">The sender info (For example Main Form)</param>
327:           /// <param name="e">Event Arguments</param>
328:           private void btnSifreUygunMu_Click(object sender, EventArgs e)
329:           {
```

```
330:                 if (string.IsNullOrEmpty(txtSifrelenecekMetin.Text.Trim()))
331:                     MessageBox.Show(Sabitler.SifreMetniGirilmedi);
332:                 else
333:                 {
334:                     lbSifreSonuc.Items.Clear();
335:                     var sb = new StringBuilder();
336:                     sb.Append("Girilen Sifrenin Uygunluk Durumu: ");
337:
sb.Append(SifrelemeIslemleri.SifreUygunMu(txtSifrelenecekMetin.Text.Trim()));
338:                     lbSifreSonuc.Items.Add(sb.ToString());
339:                 }
340:             }
341:             #endregion
342:
343:             #region Sayi Islemleri
344:             /// <summary>
345:             /// Paneldeki metin kutusuna girilen metnin sayisal bir deger olup olmadigini
346:             /// Kontrol eden ve sonucu gösteren metottur.
347:             /// </summary>
348:             /// <param name="sender">The sender info (For example Main Form)</param>
349:             /// <param name="e">Event Arguments</param>
350:             private void btnSayisalMi_Click(object sender, EventArgs e)
351:             {
352:                 if (string.IsNullOrEmpty(txtSayi.Text.Trim()))
353:                     MessageBox.Show(Sabitler.MetinGirilmedi);
354:                 else
355:                 {
356:                     lbSayiSonuc.Items.Clear();
357:                     var sb = new StringBuilder();
358:                     sb.Append("Girilen Metin Sayi Mi?: ");
359:                     sb.Append(SayiIslemleri.SayisalMi(txtSayi.Text.Trim()) ? "Evet" :
"Hayir");
360:                     lbSayiSonuc.Items.Add(sb.ToString());
361:                 }
362:             }
363:
364:             /// <summary>
365:             /// Paneldeki metin kutusuna girilen para birimi cinsinden rakamin
366:             /// Yaziya çevrilmesini saglayan ve sonucu listbox üzerinde gösteren bir
metottur.
367:             /// </summary>
368:             /// <param name="sender">The sender info (For example Main Form)</param>
369:             /// <param name="e">Event Arguments</param>
370:             private void btnMetneDonustur_Click(object sender, EventArgs e)
371:             {
372:                 if (string.IsNullOrEmpty(txtSayi.Text.Trim()))
373:                     MessageBox.Show(Sabitler.MetinGirilmedi);
374:                 else if (!SayiIslemleri.SayisalMi(txtSayi.Text.Trim().Replace(",",
".").Replace(".", "")))
375:                     MessageBox.Show(Sabitler.MetinSayisalDegil);
376:                 else
377:                 {
378:                     var metinSadeceSayiKismi = txtSayi.Text.Trim();
379:                     var sayi = SayiIslemleri.SayiHalineCevir(metinSadeceSayiKismi);
380:                     lbSayiSonuc.Items.Clear();
381:                     var sb = new StringBuilder();
382:                     sb.Append("Girilen Sayinin Metin Hali (Parasal): ");
383:                     sb.Append(SayiIslemleri.SayiyiYaziyaCevir(sayi));
384:                     lbSayiSonuc.Items.Add(sb.ToString());
385:                 }
386:             }
387:
388:             /// <summary>
389:             /// Rastgele olacak sekilde 6 haneli kod olusturmayi saglayan bir metottur.
390:             /// </summary>
391:             /// <param name="sender">The sender info (For example Main Form)</param>
392:             /// <param name="e">Event Arguments</param>
393:             private void btnOnayKoduOlustur_Click(object sender, EventArgs e)
394:             {
```

```
395:                   lbSayiSonuc.Items.Clear();
396:                   var sb = new StringBuilder();
397:                   sb.Append("Olusturulan 6 haneli onay kodu: ");
398:                   sb.Append(SayiIslemleri.AltiHaneliOnayKoduOlustur());
399:                   lbSayiSonuc.Items.Add(sb.ToString());
400:              }
401:          #endregion
402:
403:          #region Kisi Islemleri
404:          /// <summary>
405:          /// Paneldeki metin kutusuna girilen TC Kimlik numarasinin
406:          /// Standart TC Kimlik No algoritmasina uygun olup olmadigini kontrol eden bir
metottur.
407:          /// </summary>
408:          /// <param name="sender">The sender info (For example Main Form)</param>
409:          /// <param name="e">Event Arguments</param>
410:          private void btnTcKimlikDogrula_Click(object sender, EventArgs e)
411:          {
412:              if (string.IsNullOrEmpty(txtKisiMetni.Text.Trim()))
413:                  MessageBox.Show(Sabitler.MetinGirilmedi);
414:              else if (!SayiIslemleri.SayisalMi(txtKisiMetni.Text.Trim()))
415:                  MessageBox.Show(Sabitler.MetinSayisalDegil);
416:              else if (txtKisiMetni.Text.Trim().Length != 11)
417:                  MessageBox.Show(Sabitler.MetinTcKimlikNoDegil);
418:              else
419:              {
420:                  lbKisiSonuc.Items.Clear();
421:                  var sb = new StringBuilder();
422:                  sb.Append("TC Kimlik No Dogru Mu?: ");
423:                  sb.Append(KisiIslemleri.DogrulaTcKimlikNo(txtKisiMetni.Text.Trim()) ?
"Evet" : "Hayir");
424:                  lbKisiSonuc.Items.Add(sb.ToString());
425:              }
426:          }
427:
428:          /// <summary>
429:          /// Paneldeki metin kutusuna girilen IBAN Numarasinin
430:          /// Standart IBAN algoritmasina uygun olup olmadigini kontrol eden bir
metottur.
431:          /// </summary>
432:          /// <param name="sender">The sender info (For example Main Form)</param>
433:          /// <param name="e">Event Arguments</param>
434:          private void btnIbanDogrula_Click(object sender, EventArgs e)
435:          {
436:              if (string.IsNullOrEmpty(txtKisiMetni.Text.Trim()))
437:                  MessageBox.Show(Sabitler.MetinGirilmedi);
438:              else if (txtKisiMetni.Text.Trim().Length != 26)
439:                  MessageBox.Show(Sabitler.MetinIbanNoDegil +
txtKisiMetni.Text.Trim().Length);
440:              else
441:              {
442:                  lbKisiSonuc.Items.Clear();
443:                  var sb = new StringBuilder();
444:                  sb.Append("IBAN Dogru Mu?: ");
445:                  sb.Append(KisiIslemleri.DogrulaIBAN(txtKisiMetni.Text.Trim()) ? "Evet"
: "Hayir");
446:                  lbKisiSonuc.Items.Add(sb.ToString());
447:              }
448:          }
449:
450:          /// <summary>
451:          /// Paneldeki metin kutusuna girilen metnin
452:          /// Geçerli bir e-posta adresi olup olmadigini kontrol eden bir metottur.
453:          /// </summary>
454:          /// <param name="sender">The sender info (For example Main Form)</param>
455:          /// <param name="e">Event Arguments</param>
456:          private void btnEPostaDogrula_Click(object sender, EventArgs e)
457:          {
458:              if (string.IsNullOrEmpty(txtKisiMetni.Text.Trim()))
```

```
459:                           MessageBox.Show(Sabitler.MetinGirilmedi);
460:                       else
461:                       {
462:                           lbKisiSonuc.Items.Clear();
463:                           var sb = new StringBuilder();
464:                           sb.Append("E-Posta Dogru Mu?: ");
465:                           sb.Append(KisiIslemleri.DogrulaEPosta(txtKisiMetni.Text.Trim()) ?
"Evet" : "Hayir");
466:                           lbKisiSonuc.Items.Add(sb.ToString());
467:                       }
468:                   }
469:           #endregion
470:
471:           #region Metin Islemleri
472:           /// <summary>
473:           /// Paneldeki metin kutusuna girilen metnin
474:           /// Kelimelerinin ilk harflerini büyük yapmaya yarayan bir metottur.
475:           /// </summary>
476:           /// <param name="sender">The sender info (For example Main Form)</param>
477:           /// <param name="e">Event Arguments</param>
478:           private void btnIlkHarfleriBuyukYap_Click(object sender, EventArgs e)
479:           {
480:               if (string.IsNullOrEmpty(txtMetin.Text.Trim()))
481:                   MessageBox.Show(Sabitler.MetinGirilmedi);
482:               else
483:               {
484:                   lbMetinSonuc.Items.Clear();
485:                   var sb = new StringBuilder();
486:                   sb.Append("Metnin Düzenlenmis Hali: ");
487:                   sb.Append(txtMetin.Text.Trim().MetinIlkKarakterleriBuyukYap());
488:                   lbMetinSonuc.Items.Add(sb.ToString());
489:               }
490:           }
491:
492:           /// <summary>
493:           /// Paneldeki metin kutusuna girilen metnin
494:           /// Sadece ilk harfini büyük yapmaya yarayan bir metottur.
495:           /// </summary>
496:           /// <param name="sender">The sender info (For example Main Form)</param>
497:           /// <param name="e">Event Arguments</param>
498:           private void btnIlkHarfiBuyukYap_Click(object sender, EventArgs e)
499:           {
500:               if (string.IsNullOrEmpty(txtMetin.Text.Trim()))
501:                   MessageBox.Show(Sabitler.MetinGirilmedi);
502:               else
503:               {
504:                   lbMetinSonuc.Items.Clear();
505:                   var sb = new StringBuilder();
506:                   sb.Append("Metnin Düzenlenmis Hali: ");
507:
sb.Append(MetinIslemleri.MetinIlkKarakterBuyukDigerleriKucukYap(txtMetin.Text.Trim()));
508:                   lbMetinSonuc.Items.Add(sb.ToString());
509:               }
510:           }
511:
512:           /// <summary>
513:           /// Paneldeki metin kutusuna girilen metnin
514:           /// Içerisindeki Türkçe karakterleri Ingilizee hallerine çeviren bir metottur.
515:           /// </summary>
516:           /// <param name="sender">The sender info (For example Main Form)</param>
517:           /// <param name="e">Event Arguments</param>
518:           private void btnTurkceKarakterDuzelt_Click(object sender, EventArgs e)
519:           {
520:               if (string.IsNullOrEmpty(txtMetin.Text.Trim()))
521:                   MessageBox.Show(Sabitler.MetinGirilmedi);
522:               else
523:               {
524:                   lbMetinSonuc.Items.Clear();
525:                   var sb = new StringBuilder();
```

```
526:                    sb.Append("Metnin Düzenlenmis Hali: ");
527:
sb.Append(MetinIslemleri.TurkceKarakterleriDuzelt(txtMetin.Text.Trim()));
528:                    lbMetinSonuc.Items.Add(sb.ToString());
529:                }
530:            }
531:
532:          /// <summary>
533:          /// KVKK durumlarindan dolayi TC Kimlik No ve Vergi No gibi alanlarin
534:          /// Matbu çiktilarda gizlenmesi gibi bir durum söz konusu oldugundan
535:          /// Girilen bu 10 veya 11 karakterlik metinleri sifrelemeye yarayan bir
metottur
536:          /// </summary>
537:          /// <param name="sender">The sender info (For example Main Form)</param>
538:          /// <param name="e">Event Arguments</param>
539:          private void btnMetinSifrele_Click(object sender, EventArgs e)
540:          {
541:              if (string.IsNullOrEmpty(txtMetin.Text.Trim()))
542:                  MessageBox.Show(Sabitler.MetinGirilmedi);
543:              else if (!SayiIslemleri.SayisalMi(txtMetin.Text.Trim()))
544:                  MessageBox.Show(Sabitler.MetinSayisalDegil);
545:              else if (txtMetin.Text.Trim().Length != 11 && txtMetin.Text.Trim().Length
!= 10)
546:                  MessageBox.Show(Sabitler.MetinTcKimlikNoVeyaVergiNoDegil);
547:              else
548:              {
549:                  lbMetinSonuc.Items.Clear();
550:                  var sb = new StringBuilder();
551:                  sb.Append("TC Kimlik No veya Vergi No'nun Sifrelenmis Hali: ");
552:                  sb.Append(MetinIslemleri.MetinSifrele(txtMetin.Text.Trim()));
553:                  lbMetinSonuc.Items.Add(sb.ToString());
554:              }
555:          }
556:
557:          /// <summary>
558:          /// Ekrandan çesitli karakterler içeren telefon numaralari alindigi zaman
559:          /// Bu telefon numarasini olmasi gerektigi hale çevirmeye yarayan bir metottur.
560:          /// </summary>
561:          /// <param name="sender">The sender info (For example Main Form)</param>
562:          /// <param name="e">Event Arguments</param>
563:          private void btnTelefonNoDuzenle_Click(object sender, EventArgs e)
564:          {
565:              if (string.IsNullOrEmpty(txtMetin.Text.Trim()))
566:                  MessageBox.Show(Sabitler.MetinGirilmedi);
567:              else
568:              {
569:                  lbMetinSonuc.Items.Clear();
570:                  var sb = new StringBuilder();
571:                  sb.Append("Telefon Numarasinin Düzenlenmis Hali: ");
572:                  sb.Append(MetinIslemleri.TelefonNoDuzenle(txtMetin.Text.Trim()));
573:                  lbMetinSonuc.Items.Add(sb.ToString());
574:              }
575:          }
576:
577:          /// <summary>
578:          /// Ekrandan girilen metnin geçerli bir e-posta adresi olup olmadigini kontrol
eden
579:          /// Sonucu listbox'a yazdiran metottur.
580:          /// </summary>
581:          /// <param name="sender">The sender info (For example Main Form)</param>
582:          /// <param name="e">Event Arguments</param>
583:          private void btnEPostaKontrol_Click(object sender, EventArgs e)
584:          {
585:              if (string.IsNullOrEmpty(txtMetin.Text.Trim()))
586:                  MessageBox.Show(Sabitler.MetinGirilmedi);
587:              else
588:              {
589:                  lbMetinSonuc.Items.Clear();
590:                  var gecerliMi =
```

1

```
MetinIslemleri.MailAdresiGecerliMi(txtMetin.Text.Trim());
591:              lbMetinSonuc.Items.Add(txtMetin.Text.Trim());
592:              lbMetinSonuc.Items.Add(gecerliMi ? Sabitler.EPostaAdresiGecerli :
Sabitler.EPostaAdresiGecerliDegil);
593:          }
594:      }
595:      #endregion
596:  }
597: }
```

Namespaces

Name	Description
CSharpHelperMethods (⊡ see page 1)	This is namespace CSharpHelperMethods.

1.2.9 FrmAnaSayfa.Designer.cs

This is file FrmAnaSayfa.Designer.cs.

Body Source

```
 1: ?namespace CSharpHelperMethods.UI
 2: {
 3:     partial class FrmAnaSayfa
 4:     {
 5:         /// <summary>
 6:         /// Required designer variable.
 7:         /// </summary>
 8:         private System.ComponentModel.IContainer components = null;
 9:
10:         /// <summary>
11:         /// Clean up any resources being used.
12:         /// </summary>
13:         /// <param name="disposing">true if managed resources should be disposed;
otherwise, false.</param>
14:         protected override void Dispose(bool disposing)
15:         {
16:             if (disposing && (components != null))
17:             {
18:                 components.Dispose();
19:             }
20:             base.Dispose(disposing);
21:         }
22:
23:         #region Windows Form Designer generated code
24:
25:         /// <summary>
26:         /// Required method for Designer support - do not modify
27:         /// the contents of this method with the code editor.
28:         /// </summary>
29:         private void InitializeComponent()
30:         {
31:             this.cmbIslemTuru = new System.Windows.Forms.ComboBox();
32:             this.label1 = new System.Windows.Forms.Label();
33:             this.pnlTarihIslemleri = new System.Windows.Forms.Panel();
34:             this.btnAyFarki = new System.Windows.Forms.Button();
35:             this.label8 = new System.Windows.Forms.Label();
36:             this.btnTarihAraligi = new System.Windows.Forms.Button();
37:             this.btnYasHesaplaMetinsel = new System.Windows.Forms.Button();
38:             this.btnYasHesapla = new System.Windows.Forms.Button();
39:             this.label4 = new System.Windows.Forms.Label();
40:             this.lbSonuc = new System.Windows.Forms.ListBox();
41:             this.dtpBitisTarihi = new System.Windows.Forms.DateTimePicker();
42:             this.dtpBaslangicTarihi = new System.Windows.Forms.DateTimePicker();
43:             this.label3 = new System.Windows.Forms.Label();
```

```
 44:                    this.label2 = new System.Windows.Forms.Label();
 45:                    this.pnlSifreIslemleri = new System.Windows.Forms.Panel();
 46:                    this.label7 = new System.Windows.Forms.Label();
 47:                    this.btnSifreUygunMu = new System.Windows.Forms.Button();
 48:                    this.btnRastgeleSifreUret = new System.Windows.Forms.Button();
 49:                    this.btnSha512Hash = new System.Windows.Forms.Button();
 50:                    this.btnSha256Hash = new System.Windows.Forms.Button();
 51:                    this.label6 = new System.Windows.Forms.Label();
 52:                    this.lbSifreSonuc = new System.Windows.Forms.ListBox();
 53:                    this.btnMd5Hash = new System.Windows.Forms.Button();
 54:                    this.txtSifrelenecekMetin = new System.Windows.Forms.TextBox();
 55:                    this.label5 = new System.Windows.Forms.Label();
 56:                    this.pnlSayiIslemleri = new System.Windows.Forms.Panel();
 57:                    this.label9 = new System.Windows.Forms.Label();
 58:                    this.btnOnayKoduOlustur = new System.Windows.Forms.Button();
 59:                    this.btnMetneDonustur = new System.Windows.Forms.Button();
 60:                    this.label10 = new System.Windows.Forms.Label();
 61:                    this.lbSayiSonuc = new System.Windows.Forms.ListBox();
 62:                    this.btnSayisalMi = new System.Windows.Forms.Button();
 63:                    this.txtSayi = new System.Windows.Forms.TextBox();
 64:                    this.label11 = new System.Windows.Forms.Label();
 65:                    this.pnlKisiIslemleri = new System.Windows.Forms.Panel();
 66:                    this.label12 = new System.Windows.Forms.Label();
 67:                    this.btnEPostaDogrula = new System.Windows.Forms.Button();
 68:                    this.btnIbanDogrula = new System.Windows.Forms.Button();
 69:                    this.label13 = new System.Windows.Forms.Label();
 70:                    this.lbKisiSonuc = new System.Windows.Forms.ListBox();
 71:                    this.btnTcKimlikDogrula = new System.Windows.Forms.Button();
 72:                    this.txtKisiMetni = new System.Windows.Forms.TextBox();
 73:                    this.label14 = new System.Windows.Forms.Label();
 74:                    this.pnlMetinIslemleri = new System.Windows.Forms.Panel();
 75:                    this.btnTelefonNoDuzenle = new System.Windows.Forms.Button();
 76:                    this.btnMetinSifrele = new System.Windows.Forms.Button();
 77:                    this.label15 = new System.Windows.Forms.Label();
 78:                    this.btnTurkceKarakterDuzelt = new System.Windows.Forms.Button();
 79:                    this.btnIlkHarfiBuyukYap = new System.Windows.Forms.Button();
 80:                    this.label16 = new System.Windows.Forms.Label();
 81:                    this.lbMetinSonuc = new System.Windows.Forms.ListBox();
 82:                    this.btnIlkHarfleriBuyukYap = new System.Windows.Forms.Button();
 83:                    this.txtMetin = new System.Windows.Forms.TextBox();
 84:                    this.label17 = new System.Windows.Forms.Label();
 85:                    this.btnEPostaKontrol = new System.Windows.Forms.Button();
 86:              this.pnlTarihIslemleri.SuspendLayout();
 87:              this.pnlSifreIslemleri.SuspendLayout();
 88:              this.pnlSayiIslemleri.SuspendLayout();
 89:              this.pnlKisiIslemleri.SuspendLayout();
 90:              this.pnlMetinIslemleri.SuspendLayout();
 91:              this.SuspendLayout();
 92:              //
 93:              // cmbIslemTuru
 94:              //
 95:              this.cmbIslemTuru.FormattingEnabled = true;
 96:              this.cmbIslemTuru.Items.AddRange(new object[] {"Metin Islemleri", "Tarih
Islemleri", "Sayi Islemleri", "Sifre Islemleri", "Kisi Islemleri"});
 97:              this.cmbIslemTuru.Location = new System.Drawing.Point(176, 16);
 98:              this.cmbIslemTuru.Margin = new System.Windows.Forms.Padding(4, 4, 4, 4);
 99:              this.cmbIslemTuru.Name = "cmbIslemTuru";
100:              this.cmbIslemTuru.Size = new System.Drawing.Size(265, 24);
101:              this.cmbIslemTuru.TabIndex = 0;
102:              this.cmbIslemTuru.SelectedIndexChanged += new
System.EventHandler(this.cmbIslemTuru_SelectedIndexChanged);
103:              //
104:              // label1
105:              //
106:              this.label1.AutoSize = true;
107:              this.label1.Location = new System.Drawing.Point(16, 20);
108:              this.label1.Margin = new System.Windows.Forms.Padding(4, 0, 4, 0);
109:              this.label1.Name = "label1";
110:              this.label1.Size = new System.Drawing.Size(123, 17);
```

```
111:                    this.label1.TabIndex = 1;
112:                    this.label1.Text = "Islem Türü Seçiniz";
113:                    //
114:                    // pnlTarihIslemleri
115:                    //
116:                    this.pnlTarihIslemleri.BorderStyle =
System.Windows.Forms.BorderStyle.FixedSingle;
117:                    this.pnlTarihIslemleri.Controls.Add(this.btnAyFarki);
118:                    this.pnlTarihIslemleri.Controls.Add(this.label8);
119:                    this.pnlTarihIslemleri.Controls.Add(this.btnTarihAraligi);
120:                    this.pnlTarihIslemleri.Controls.Add(this.btnYasHesaplaMetinsel);
121:                    this.pnlTarihIslemleri.Controls.Add(this.btnYasHesapla);
122:                    this.pnlTarihIslemleri.Controls.Add(this.label4);
123:                    this.pnlTarihIslemleri.Controls.Add(this.lbSonuc);
124:                    this.pnlTarihIslemleri.Controls.Add(this.dtpBitisTarihi);
125:                    this.pnlTarihIslemleri.Controls.Add(this.dtpBaslangicTarihi);
126:                    this.pnlTarihIslemleri.Controls.Add(this.label3);
127:                    this.pnlTarihIslemleri.Controls.Add(this.label2);
128:                    this.pnlTarihIslemleri.Location = new System.Drawing.Point(20, 69);
129:                    this.pnlTarihIslemleri.Margin = new System.Windows.Forms.Padding(4, 4, 4,
4);
130:                    this.pnlTarihIslemleri.Name = "pnlTarihIslemleri";
131:                    this.pnlTarihIslemleri.Size = new System.Drawing.Size(498, 222);
132:                    this.pnlTarihIslemleri.TabIndex = 2;
133:                    //
134:                    // btnAyFarki
135:                    //
136:                    this.btnAyFarki.Location = new System.Drawing.Point(381, 94);
137:                    this.btnAyFarki.Margin = new System.Windows.Forms.Padding(4, 4, 4, 4);
138:                    this.btnAyFarki.Name = "btnAyFarki";
139:                    this.btnAyFarki.Size = new System.Drawing.Size(105, 28);
140:                    this.btnAyFarki.TabIndex = 12;
141:                    this.btnAyFarki.Text = "Ay Farki";
142:                    this.btnAyFarki.UseVisualStyleBackColor = true;
143:                    this.btnAyFarki.Click += new System.EventHandler(this.btnAyFarki_Click);
144:                    //
145:                    // label8
146:                    //
147:                    this.label8.AutoSize = true;
148:                    this.label8.Dock = System.Windows.Forms.DockStyle.Top;
149:                    this.label8.Location = new System.Drawing.Point(0, 0);
150:                    this.label8.Margin = new System.Windows.Forms.Padding(4, 0, 4, 0);
151:                    this.label8.Name = "label8";
152:                    this.label8.Size = new System.Drawing.Size(96, 17);
153:                    this.label8.TabIndex = 11;
154:                    this.label8.Text = "Tarih Islemleri";
155:                    //
156:                    // btnTarihAraligi
157:                    //
158:                    this.btnTarihAraligi.Location = new System.Drawing.Point(272, 94);
159:                    this.btnTarihAraligi.Margin = new System.Windows.Forms.Padding(4, 4, 4, 4);
160:                    this.btnTarihAraligi.Name = "btnTarihAraligi";
161:                    this.btnTarihAraligi.Size = new System.Drawing.Size(101, 28);
162:                    this.btnTarihAraligi.TabIndex = 10;
163:                    this.btnTarihAraligi.Text = "Tarih Araligi";
164:                    this.btnTarihAraligi.UseVisualStyleBackColor = true;
165:                    this.btnTarihAraligi.Click += new
System.EventHandler(this.btnTarihAraligi_Click);
166:                    //
167:                    // btnYasHesaplaMetinsel
168:                    //
169:                    this.btnYasHesaplaMetinsel.Location = new System.Drawing.Point(112, 94);
170:                    this.btnYasHesaplaMetinsel.Margin = new System.Windows.Forms.Padding(4, 4,
4, 4);
171:                    this.btnYasHesaplaMetinsel.Name = "btnYasHesaplaMetinsel";
172:                    this.btnYasHesaplaMetinsel.Size = new System.Drawing.Size(151, 28);
173:                    this.btnYasHesaplaMetinsel.TabIndex = 9;
174:                    this.btnYasHesaplaMetinsel.Text = "Yas Hesapla (Metin)";
175:                    this.btnYasHesaplaMetinsel.UseVisualStyleBackColor = true;
```

```
176:                    this.btnYasHesaplaMetinsel.Click += new
System.EventHandler(this.btnYasHesaplaMetinsel_Click);
177:                    //
178:                    // btnYasHesapla
179:                    //
180:                    this.btnYasHesapla.Location = new System.Drawing.Point(4, 94);
181:                    this.btnYasHesapla.Margin = new System.Windows.Forms.Padding(4, 4, 4, 4);
182:                    this.btnYasHesapla.Name = "btnYasHesapla";
183:                    this.btnYasHesapla.Size = new System.Drawing.Size(100, 28);
184:                    this.btnYasHesapla.TabIndex = 8;
185:                    this.btnYasHesapla.Text = "Yas Hesapla";
186:                    this.btnYasHesapla.UseVisualStyleBackColor = true;
187:                    this.btnYasHesapla.Click += new
System.EventHandler(this.btnYasHesapla_Click);
188:                    //
189:                    // label4
190:                    //
191:                    this.label4.AutoSize = true;
192:                    this.label4.Location = new System.Drawing.Point(24, 134);
193:                    this.label4.Margin = new System.Windows.Forms.Padding(4, 0, 4, 0);
194:                    this.label4.Name = "label4";
195:                    this.label4.Size = new System.Drawing.Size(48, 17);
196:                    this.label4.TabIndex = 7;
197:                    this.label4.Text = "Sonuç";
198:                    //
199:                    // lbSonuc
200:                    //
201:                    this.lbSonuc.FormattingEnabled = true;
202:                    this.lbSonuc.HorizontalScrollbar = true;
203:                    this.lbSonuc.ItemHeight = 16;
204:                    this.lbSonuc.Location = new System.Drawing.Point(83, 129);
205:                    this.lbSonuc.Margin = new System.Windows.Forms.Padding(4, 4, 4, 4);
206:                    this.lbSonuc.Name = "lbSonuc";
207:                    this.lbSonuc.Size = new System.Drawing.Size(340, 84);
208:                    this.lbSonuc.TabIndex = 6;
209:                    //
210:                    // dtpBitisTarihi
211:                    //
212:                    this.dtpBitisTarihi.Format =
System.Windows.Forms.DateTimePickerFormat.Short;
213:                    this.dtpBitisTarihi.Location = new System.Drawing.Point(156, 53);
214:                    this.dtpBitisTarihi.Margin = new System.Windows.Forms.Padding(4, 4, 4, 4);
215:                    this.dtpBitisTarihi.Name = "dtpBitisTarihi";
216:                    this.dtpBitisTarihi.ShowCheckBox = true;
217:                    this.dtpBitisTarihi.Size = new System.Drawing.Size(265, 22);
218:                    this.dtpBitisTarihi.TabIndex = 5;
219:                    //
220:                    // dtpBaslangicTarihi
221:                    //
222:                    this.dtpBaslangicTarihi.Format =
System.Windows.Forms.DateTimePickerFormat.Short;
223:                    this.dtpBaslangicTarihi.Location = new System.Drawing.Point(156, 21);
224:                    this.dtpBaslangicTarihi.Margin = new System.Windows.Forms.Padding(4, 4, 4,
4);
225:                    this.dtpBaslangicTarihi.Name = "dtpBaslangicTarihi";
226:                    this.dtpBaslangicTarihi.ShowCheckBox = true;
227:                    this.dtpBaslangicTarihi.Size = new System.Drawing.Size(265, 22);
228:                    this.dtpBaslangicTarihi.TabIndex = 4;
229:                    //
230:                    // label3
231:                    //
232:                    this.label3.AutoSize = true;
233:                    this.label3.Location = new System.Drawing.Point(59, 58);
234:                    this.label3.Margin = new System.Windows.Forms.Padding(4, 0, 4, 0);
235:                    this.label3.Name = "label3";
236:                    this.label3.Size = new System.Drawing.Size(74, 17);
237:                    this.label3.TabIndex = 3;
238:                    this.label3.Text = "Bitis Tarihi";
239:                    //
```

1

```
240:                // label2
241:                //
242:                this.label2.AutoSize = true;
243:                this.label2.Location = new System.Drawing.Point(23, 30);
244:                this.label2.Margin = new System.Windows.Forms.Padding(4, 0, 4, 0);
245:                this.label2.Name = "label2";
246:                this.label2.Size = new System.Drawing.Size(109, 17);
247:                this.label2.TabIndex = 2;
248:                this.label2.Text = "Baslangiç Tarihi";
249:                //
250:                // pnlSifreIslemleri
251:                //
252:                this.pnlSifreIslemleri.BorderStyle =
System.Windows.Forms.BorderStyle.FixedSingle;
253:                this.pnlSifreIslemleri.Controls.Add(this.label7);
254:                this.pnlSifreIslemleri.Controls.Add(this.btnSifreUygunMu);
255:                this.pnlSifreIslemleri.Controls.Add(this.btnRastgeleSifreUret);
256:                this.pnlSifreIslemleri.Controls.Add(this.btnSha512Hash);
257:                this.pnlSifreIslemleri.Controls.Add(this.btnSha256Hash);
258:                this.pnlSifreIslemleri.Controls.Add(this.label6);
259:                this.pnlSifreIslemleri.Controls.Add(this.lbSifreSonuc);
260:                this.pnlSifreIslemleri.Controls.Add(this.btnMd5Hash);
261:                this.pnlSifreIslemleri.Controls.Add(this.txtSifrelenecekMetin);
262:                this.pnlSifreIslemleri.Controls.Add(this.label5);
263:                this.pnlSifreIslemleri.Location = new System.Drawing.Point(548, 69);
264:                this.pnlSifreIslemleri.Margin = new System.Windows.Forms.Padding(4, 4, 4,
4);
265:                this.pnlSifreIslemleri.Name = "pnlSifreIslemleri";
266:                this.pnlSifreIslemleri.Size = new System.Drawing.Size(502, 222);
267:                this.pnlSifreIslemleri.TabIndex = 3;
268:                //
269:                // label7
270:                //
271:                this.label7.AutoSize = true;
272:                this.label7.Dock = System.Windows.Forms.DockStyle.Top;
273:                this.label7.Location = new System.Drawing.Point(0, 0);
274:                this.label7.Margin = new System.Windows.Forms.Padding(4, 0, 4, 0);
275:                this.label7.Name = "label7";
276:                this.label7.Size = new System.Drawing.Size(92, 17);
277:                this.label7.TabIndex = 18;
278:                this.label7.Text = "Sifre Islemleri";
279:                this.label7.TextAlign = System.Drawing.ContentAlignment.TopCenter;
280:                //
281:                // btnSifreUygunMu
282:                //
283:                this.btnSifreUygunMu.Location = new System.Drawing.Point(231, 87);
284:                this.btnSifreUygunMu.Margin = new System.Windows.Forms.Padding(4, 4, 4, 4);
285:                this.btnSifreUygunMu.Name = "btnSifreUygunMu";
286:                this.btnSifreUygunMu.Size = new System.Drawing.Size(193, 28);
287:                this.btnSifreUygunMu.TabIndex = 17;
288:                this.btnSifreUygunMu.Text = "Sifre Uygun Mu?";
289:                this.btnSifreUygunMu.UseVisualStyleBackColor = true;
290:                this.btnSifreUygunMu.Click += new
System.EventHandler(this.btnSifreUygunMu_Click);
291:                //
292:                // btnRastgeleSifreUret
293:                //
294:                this.btnRastgeleSifreUret.Location = new System.Drawing.Point(28, 87);
295:                this.btnRastgeleSifreUret.Margin = new System.Windows.Forms.Padding(4, 4,
4, 4);
296:                this.btnRastgeleSifreUret.Name = "btnRastgeleSifreUret";
297:                this.btnRastgeleSifreUret.Size = new System.Drawing.Size(193, 28);
298:                this.btnRastgeleSifreUret.TabIndex = 16;
299:                this.btnRastgeleSifreUret.Text = "Rastgele Sifre Üret";
300:                this.btnRastgeleSifreUret.UseVisualStyleBackColor = true;
301:                this.btnRastgeleSifreUret.Click += new
System.EventHandler(this.btnRastgeleSifreUret_Click);
302:                //
303:                // btnSha512Hash
```

```
304:               //
305:               this.btnSha512Hash.Location = new System.Drawing.Point(324, 52);
306:               this.btnSha512Hash.Margin = new System.Windows.Forms.Padding(4, 4, 4, 4);
307:               this.btnSha512Hash.Name = "btnSha512Hash";
308:               this.btnSha512Hash.Size = new System.Drawing.Size(100, 28);
309:               this.btnSha512Hash.TabIndex = 15;
310:               this.btnSha512Hash.Text = "SHA 512";
311:               this.btnSha512Hash.UseVisualStyleBackColor = true;
312:               this.btnSha512Hash.Click += new
System.EventHandler(this.btnSha512Hash_Click);
313:               //
314:               // btnSha256Hash
315:               //
316:               this.btnSha256Hash.Location = new System.Drawing.Point(175, 52);
317:               this.btnSha256Hash.Margin = new System.Windows.Forms.Padding(4, 4, 4, 4);
318:               this.btnSha256Hash.Name = "btnSha256Hash";
319:               this.btnSha256Hash.Size = new System.Drawing.Size(100, 28);
320:               this.btnSha256Hash.TabIndex = 14;
321:               this.btnSha256Hash.Text = "SHA 256";
322:               this.btnSha256Hash.UseVisualStyleBackColor = true;
323:               this.btnSha256Hash.Click += new
System.EventHandler(this.btnSha256Hash_Click);
324:               //
325:               // label6
326:               //
327:               this.label6.AutoSize = true;
328:               this.label6.Location = new System.Drawing.Point(28, 123);
329:               this.label6.Margin = new System.Windows.Forms.Padding(4, 0, 4, 0);
330:               this.label6.Name = "label6";
331:               this.label6.Size = new System.Drawing.Size(48, 17);
332:               this.label6.TabIndex = 13;
333:               this.label6.Text = "Sonuç";
334:               //
335:               // lbSifreSonuc
336:               //
337:               this.lbSifreSonuc.FormattingEnabled = true;
338:               this.lbSifreSonuc.HorizontalScrollbar = true;
339:               this.lbSifreSonuc.ItemHeight = 16;
340:               this.lbSifreSonuc.Location = new System.Drawing.Point(87, 123);
341:               this.lbSifreSonuc.Margin = new System.Windows.Forms.Padding(4, 4, 4, 4);
342:               this.lbSifreSonuc.Name = "lbSifreSonuc";
343:               this.lbSifreSonuc.Size = new System.Drawing.Size(333, 84);
344:               this.lbSifreSonuc.TabIndex = 12;
345:               //
346:               // btnMd5Hash
347:               //
348:               this.btnMd5Hash.Location = new System.Drawing.Point(28, 52);
349:               this.btnMd5Hash.Margin = new System.Windows.Forms.Padding(4, 4, 4, 4);
350:               this.btnMd5Hash.Name = "btnMd5Hash";
351:               this.btnMd5Hash.Size = new System.Drawing.Size(100, 28);
352:               this.btnMd5Hash.TabIndex = 11;
353:               this.btnMd5Hash.Text = "MD5";
354:               this.btnMd5Hash.UseVisualStyleBackColor = true;
355:               this.btnMd5Hash.Click += new System.EventHandler(this.btnMd5Hash_Click);
356:               //
357:               // txtSifrelenecekMetin
358:               //
359:               this.txtSifrelenecekMetin.Location = new System.Drawing.Point(157, 20);
360:               this.txtSifrelenecekMetin.Margin = new System.Windows.Forms.Padding(4, 4,
4, 4);
361:               this.txtSifrelenecekMetin.Name = "txtSifrelenecekMetin";
362:               this.txtSifrelenecekMetin.Size = new System.Drawing.Size(264, 22);
363:               this.txtSifrelenecekMetin.TabIndex = 4;
364:               //
365:               // label5
366:               //
367:               this.label5.AutoSize = true;
368:               this.label5.Location = new System.Drawing.Point(23, 23);
369:               this.label5.Margin = new System.Windows.Forms.Padding(4, 0, 4, 0);
```

```
370:                this.label5.Name = "label5";
371:                this.label5.Size = new System.Drawing.Size(124, 17);
372:                this.label5.TabIndex = 3;
373:                this.label5.Text = "Sifrelenecek Metin";
374:                //
375:                // pnlSayiIslemleri
376:                //
377:                this.pnlSayiIslemleri.BorderStyle =
System.Windows.Forms.BorderStyle.FixedSingle;
378:                this.pnlSayiIslemleri.Controls.Add(this.label9);
379:                this.pnlSayiIslemleri.Controls.Add(this.btnOnayKoduOlustur);
380:                this.pnlSayiIslemleri.Controls.Add(this.btnMetneDonustur);
381:                this.pnlSayiIslemleri.Controls.Add(this.label10);
382:                this.pnlSayiIslemleri.Controls.Add(this.lbSayiSonuc);
383:                this.pnlSayiIslemleri.Controls.Add(this.btnSayisalMi);
384:                this.pnlSayiIslemleri.Controls.Add(this.txtSayi);
385:                this.pnlSayiIslemleri.Controls.Add(this.label11);
386:                this.pnlSayiIslemleri.Location = new System.Drawing.Point(20, 316);
387:                this.pnlSayiIslemleri.Margin = new System.Windows.Forms.Padding(4, 4, 4,
4);
388:                this.pnlSayiIslemleri.Name = "pnlSayiIslemleri";
389:                this.pnlSayiIslemleri.Size = new System.Drawing.Size(502, 222);
390:                this.pnlSayiIslemleri.TabIndex = 19;
391:                //
392:                // label9
393:                //
394:                this.label9.AutoSize = true;
395:                this.label9.Dock = System.Windows.Forms.DockStyle.Top;
396:                this.label9.Location = new System.Drawing.Point(0, 0);
397:                this.label9.Margin = new System.Windows.Forms.Padding(4, 0, 4, 0);
398:                this.label9.Name = "label9";
399:                this.label9.Size = new System.Drawing.Size(90, 17);
400:                this.label9.TabIndex = 18;
401:                this.label9.Text = "Sayi Islemleri";
402:                this.label9.TextAlign = System.Drawing.ContentAlignment.TopCenter;
403:                //
404:                // btnOnayKoduOlustur
405:                //
406:                this.btnOnayKoduOlustur.Location = new System.Drawing.Point(324, 52);
407:                this.btnOnayKoduOlustur.Margin = new System.Windows.Forms.Padding(4, 4, 4,
4);
408:                this.btnOnayKoduOlustur.Name = "btnOnayKoduOlustur";
409:                this.btnOnayKoduOlustur.Size = new System.Drawing.Size(100, 28);
410:                this.btnOnayKoduOlustur.TabIndex = 15;
411:                this.btnOnayKoduOlustur.Text = "Onay Kodu";
412:                this.btnOnayKoduOlustur.UseVisualStyleBackColor = true;
413:                this.btnOnayKoduOlustur.Click += new
System.EventHandler(this.btnOnayKoduOlustur_Click);
414:                //
415:                // btnMetneDonustur
416:                //
417:                this.btnMetneDonustur.Location = new System.Drawing.Point(136, 52);
418:                this.btnMetneDonustur.Margin = new System.Windows.Forms.Padding(4, 4, 4,
4);
419:                this.btnMetneDonustur.Name = "btnMetneDonustur";
420:                this.btnMetneDonustur.Size = new System.Drawing.Size(180, 28);
421:                this.btnMetneDonustur.TabIndex = 14;
422:                this.btnMetneDonustur.Text = "Para Metnine Dönüştür";
423:                this.btnMetneDonustur.UseVisualStyleBackColor = true;
424:                this.btnMetneDonustur.Click += new
System.EventHandler(this.btnMetneDonustur_Click);
425:                //
426:                // label10
427:                //
428:                this.label10.AutoSize = true;
429:                this.label10.Location = new System.Drawing.Point(24, 96);
430:                this.label10.Margin = new System.Windows.Forms.Padding(4, 0, 4, 0);
431:                this.label10.Name = "label10";
432:                this.label10.Size = new System.Drawing.Size(48, 17);
```

```
433:                    this.label10.TabIndex = 13;
434:                    this.label10.Text = "Sonuç";
435:                    //
436:                    // lbSayiSonuc
437:                    //
438:                    this.lbSayiSonuc.FormattingEnabled = true;
439:                    this.lbSayiSonuc.HorizontalScrollbar = true;
440:                    this.lbSayiSonuc.ItemHeight = 16;
441:                    this.lbSayiSonuc.Location = new System.Drawing.Point(83, 96);
442:                    this.lbSayiSonuc.Margin = new System.Windows.Forms.Padding(4, 4, 4, 4);
443:                    this.lbSayiSonuc.Name = "lbSayiSonuc";
444:                    this.lbSayiSonuc.Size = new System.Drawing.Size(337, 100);
445:                    this.lbSayiSonuc.TabIndex = 12;
446:                    //
447:                    // btnSayisalMi
448:                    //
449:                    this.btnSayisalMi.Location = new System.Drawing.Point(28, 52);
450:                    this.btnSayisalMi.Margin = new System.Windows.Forms.Padding(4, 4, 4, 4);
451:                    this.btnSayisalMi.Name = "btnSayisalMi";
452:                    this.btnSayisalMi.Size = new System.Drawing.Size(100, 28);
453:                    this.btnSayisalMi.TabIndex = 11;
454:                    this.btnSayisalMi.Text = "Sayisal Mi?";
455:                    this.btnSayisalMi.UseVisualStyleBackColor = true;
456:                    this.btnSayisalMi.Click += new
System.EventHandler(this.btnSayisalMi_Click);
457:                    //
458:                    // txtSayi
459:                    //
460:                    this.txtSayi.Location = new System.Drawing.Point(157, 20);
461:                    this.txtSayi.Margin = new System.Windows.Forms.Padding(4, 4, 4, 4);
462:                    this.txtSayi.Name = "txtSayi";
463:                    this.txtSayi.Size = new System.Drawing.Size(264, 22);
464:                    this.txtSayi.TabIndex = 4;
465:                    //
466:                    // label11
467:                    //
468:                    this.label11.AutoSize = true;
469:                    this.label11.Location = new System.Drawing.Point(23, 23);
470:                    this.label11.Margin = new System.Windows.Forms.Padding(4, 0, 4, 0);
471:                    this.label11.Name = "label11";
472:                    this.label11.Size = new System.Drawing.Size(124, 17);
473:                    this.label11.TabIndex = 3;
474:                    this.label11.Text = "Sifrelenecek Metin";
475:                    //
476:                    // pnlKisiIslemleri
477:                    //
478:                    this.pnlKisiIslemleri.BorderStyle =
System.Windows.Forms.BorderStyle.FixedSingle;
479:                    this.pnlKisiIslemleri.Controls.Add(this.label12);
480:                    this.pnlKisiIslemleri.Controls.Add(this.btnEPostaDogrula);
481:                    this.pnlKisiIslemleri.Controls.Add(this.btnIbanDogrula);
482:                    this.pnlKisiIslemleri.Controls.Add(this.label13);
483:                    this.pnlKisiIslemleri.Controls.Add(this.lbKisiSonuc);
484:                    this.pnlKisiIslemleri.Controls.Add(this.btnTcKimlikDogrula);
485:                    this.pnlKisiIslemleri.Controls.Add(this.txtKisiMetni);
486:                    this.pnlKisiIslemleri.Controls.Add(this.label14);
487:                    this.pnlKisiIslemleri.Location = new System.Drawing.Point(548, 316);
488:                    this.pnlKisiIslemleri.Margin = new System.Windows.Forms.Padding(4, 4, 4,
4);
489:                    this.pnlKisiIslemleri.Name = "pnlKisiIslemleri";
490:                    this.pnlKisiIslemleri.Size = new System.Drawing.Size(502, 222);
491:                    this.pnlKisiIslemleri.TabIndex = 20;
492:                    //
493:                    // label12
494:                    //
495:                    this.label12.AutoSize = true;
496:                    this.label12.Dock = System.Windows.Forms.DockStyle.Top;
497:                    this.label12.Location = new System.Drawing.Point(0, 0);
498:                    this.label12.Margin = new System.Windows.Forms.Padding(4, 0, 4, 0);
```

```
499:                    this.label12.Name = "label12";
500:                    this.label12.Size = new System.Drawing.Size(85, 17);
501:                    this.label12.TabIndex = 18;
502:                    this.label12.Text = "Kisi Islemleri";
503:                    this.label12.TextAlign = System.Drawing.ContentAlignment.TopCenter;
504:                    //
505:                    // btnEPostaDogrula
506:                    //
507:                    this.btnEPostaDogrula.Location = new System.Drawing.Point(297, 52);
508:                    this.btnEPostaDogrula.Margin = new System.Windows.Forms.Padding(4, 4, 4,
4);
509:                    this.btnEPostaDogrula.Name = "btnEPostaDogrula";
510:                    this.btnEPostaDogrula.Size = new System.Drawing.Size(127, 28);
511:                    this.btnEPostaDogrula.TabIndex = 15;
512:                    this.btnEPostaDogrula.Text = "E-Posta Dogrula";
513:                    this.btnEPostaDogrula.UseVisualStyleBackColor = true;
514:                    this.btnEPostaDogrula.Click += new
System.EventHandler(this.btnEPostaDogrula_Click);
515:                    //
516:                    // btnIbanDogrula
517:                    //
518:                    this.btnIbanDogrula.Location = new System.Drawing.Point(175, 52);
519:                    this.btnIbanDogrula.Margin = new System.Windows.Forms.Padding(4, 4, 4, 4);
520:                    this.btnIbanDogrula.Name = "btnIbanDogrula";
521:                    this.btnIbanDogrula.Size = new System.Drawing.Size(115, 28);
522:                    this.btnIbanDogrula.TabIndex = 14;
523:                    this.btnIbanDogrula.Text = "IBAN Dogrula";
524:                    this.btnIbanDogrula.UseVisualStyleBackColor = true;
525:                    this.btnIbanDogrula.Click += new
System.EventHandler(this.btnIbanDogrula_Click);
526:                    //
527:                    // label13
528:                    //
529:                    this.label13.AutoSize = true;
530:                    this.label13.Location = new System.Drawing.Point(24, 96);
531:                    this.label13.Margin = new System.Windows.Forms.Padding(4, 0, 4, 0);
532:                    this.label13.Name = "label13";
533:                    this.label13.Size = new System.Drawing.Size(48, 17);
534:                    this.label13.TabIndex = 13;
535:                    this.label13.Text = "Sonuç";
536:                    //
537:                    // lbKisiSonuc
538:                    //
539:                    this.lbKisiSonuc.FormattingEnabled = true;
540:                    this.lbKisiSonuc.HorizontalScrollbar = true;
541:                    this.lbKisiSonuc.ItemHeight = 16;
542:                    this.lbKisiSonuc.Location = new System.Drawing.Point(83, 96);
543:                    this.lbKisiSonuc.Margin = new System.Windows.Forms.Padding(4, 4, 4, 4);
544:                    this.lbKisiSonuc.Name = "lbKisiSonuc";
545:                    this.lbKisiSonuc.Size = new System.Drawing.Size(337, 100);
546:                    this.lbKisiSonuc.TabIndex = 12;
547:                    //
548:                    // btnTcKimlikDogrula
549:                    //
550:                    this.btnTcKimlikDogrula.Location = new System.Drawing.Point(28, 52);
551:                    this.btnTcKimlikDogrula.Margin = new System.Windows.Forms.Padding(4, 4, 4,
4);
552:                    this.btnTcKimlikDogrula.Name = "btnTcKimlikDogrula";
553:                    this.btnTcKimlikDogrula.Size = new System.Drawing.Size(139, 28);
554:                    this.btnTcKimlikDogrula.TabIndex = 11;
555:                    this.btnTcKimlikDogrula.Text = "TC Kimlik Dogrula";
556:                    this.btnTcKimlikDogrula.UseVisualStyleBackColor = true;
557:                    this.btnTcKimlikDogrula.Click += new
System.EventHandler(this.btnTcKimlikDogrula_Click);
558:                    //
559:                    // txtKisiMetni
560:                    //
561:                    this.txtKisiMetni.Location = new System.Drawing.Point(157, 20);
562:                    this.txtKisiMetni.Margin = new System.Windows.Forms.Padding(4, 4, 4, 4);
```

```
563:                    this.txtKisiMetni.Name = "txtKisiMetni";
564:                    this.txtKisiMetni.Size = new System.Drawing.Size(264, 22);
565:                    this.txtKisiMetni.TabIndex = 4;
566:                    //
567:                    // label14
568:                    //
569:                    this.label14.AutoSize = true;
570:                    this.label14.Location = new System.Drawing.Point(23, 23);
571:                    this.label14.Margin = new System.Windows.Forms.Padding(4, 0, 4, 0);
572:                    this.label14.Name = "label14";
573:                    this.label14.Size = new System.Drawing.Size(42, 17);
574:                    this.label14.TabIndex = 3;
575:                    this.label14.Text = "Metin";
576:                    //
577:                    // pnlMetinIslemleri
578:                    //
579:                    this.pnlMetinIslemleri.BorderStyle =
System.Windows.Forms.BorderStyle.FixedSingle;
580:                    this.pnlMetinIslemleri.Controls.Add(this.btnEPostaKontrol);
581:                    this.pnlMetinIslemleri.Controls.Add(this.btnTelefonNoDuzenle);
582:                    this.pnlMetinIslemleri.Controls.Add(this.btnMetinSifrele);
583:                    this.pnlMetinIslemleri.Controls.Add(this.label15);
584:                    this.pnlMetinIslemleri.Controls.Add(this.btnTurkceKarakterDuzelt);
585:                    this.pnlMetinIslemleri.Controls.Add(this.btnIlkHarfiBuyukYap);
586:                    this.pnlMetinIslemleri.Controls.Add(this.label16);
587:                    this.pnlMetinIslemleri.Controls.Add(this.lbMetinSonuc);
588:                    this.pnlMetinIslemleri.Controls.Add(this.btnIlkHarfleriBuyukYap);
589:                    this.pnlMetinIslemleri.Controls.Add(this.txtMetin);
590:                    this.pnlMetinIslemleri.Controls.Add(this.label17);
591:                    this.pnlMetinIslemleri.Location = new System.Drawing.Point(20, 564);
592:                    this.pnlMetinIslemleri.Margin = new System.Windows.Forms.Padding(4, 4, 4,
4);
593:                    this.pnlMetinIslemleri.Name = "pnlMetinIslemleri";
594:                    this.pnlMetinIslemleri.Size = new System.Drawing.Size(1030, 222);
595:                    this.pnlMetinIslemleri.TabIndex = 21;
596:                    //
597:                    // btnTelefonNoDuzenle
598:                    //
599:                    this.btnTelefonNoDuzenle.Location = new System.Drawing.Point(671, 52);
600:                    this.btnTelefonNoDuzenle.Margin = new System.Windows.Forms.Padding(4, 4,
4, 4);
601:                    this.btnTelefonNoDuzenle.Name = "btnTelefonNoDuzenle";
602:                    this.btnTelefonNoDuzenle.Size = new System.Drawing.Size(179, 28);
603:                    this.btnTelefonNoDuzenle.TabIndex = 20;
604:                    this.btnTelefonNoDuzenle.Text = "Telefon No Düzenle";
605:                    this.btnTelefonNoDuzenle.UseVisualStyleBackColor = true;
606:                    this.btnTelefonNoDuzenle.Click += new
System.EventHandler(this.btnTelefonNoDuzenle_Click);
607:                    //
608:                    // btnMetinSifrele
609:                    //
610:                    this.btnMetinSifrele.Location = new System.Drawing.Point(484, 52);
611:                    this.btnMetinSifrele.Margin = new System.Windows.Forms.Padding(4, 4, 4, 4);
612:                    this.btnMetinSifrele.Name = "btnMetinSifrele";
613:                    this.btnMetinSifrele.Size = new System.Drawing.Size(179, 28);
614:                    this.btnMetinSifrele.TabIndex = 19;
615:                    this.btnMetinSifrele.Text = "Metin Sifrele";
616:                    this.btnMetinSifrele.UseVisualStyleBackColor = true;
617:                    this.btnMetinSifrele.Click += new
System.EventHandler(this.btnMetinSifrele_Click);
618:                    //
619:                    // label15
620:                    //
621:                    this.label15.AutoSize = true;
622:                    this.label15.Dock = System.Windows.Forms.DockStyle.Top;
623:                    this.label15.Location = new System.Drawing.Point(0, 0);
624:                    this.label15.Margin = new System.Windows.Forms.Padding(4, 0, 4, 0);
625:                    this.label15.Name = "label15";
626:                    this.label15.Size = new System.Drawing.Size(97, 17);
```

```
627:                    this.label15.TabIndex = 18;
628:                    this.label15.Text = "Metin Islemleri";
629:                    this.label15.TextAlign = System.Drawing.ContentAlignment.TopCenter;
630:                    //
631:                    // btnTurkceKarakterDuzelt
632:                    //
633:                    this.btnTurkceKarakterDuzelt.Location = new System.Drawing.Point(297, 52);
634:                    this.btnTurkceKarakterDuzelt.Margin = new System.Windows.Forms.Padding(4,
4, 4, 4);
635:                    this.btnTurkceKarakterDuzelt.Name = "btnTurkceKarakterDuzelt";
636:                    this.btnTurkceKarakterDuzelt.Size = new System.Drawing.Size(179, 28);
637:                    this.btnTurkceKarakterDuzelt.TabIndex = 15;
638:                    this.btnTurkceKarakterDuzelt.Text = "Türkçe Karakter Düzelt";
639:                    this.btnTurkceKarakterDuzelt.UseVisualStyleBackColor = true;
640:                    this.btnTurkceKarakterDuzelt.Click += new
System.EventHandler(this.btnTurkceKarakterDuzelt_Click);
641:                    //
642:                    // btnIlkHarfiBuyukYap
643:                    //
644:                    this.btnIlkHarfiBuyukYap.Location = new System.Drawing.Point(175, 52);
645:                    this.btnIlkHarfiBuyukYap.Margin = new System.Windows.Forms.Padding(4, 4,
4, 4);
646:                    this.btnIlkHarfiBuyukYap.Name = "btnIlkHarfiBuyukYap";
647:                    this.btnIlkHarfiBuyukYap.Size = new System.Drawing.Size(115, 28);
648:                    this.btnIlkHarfiBuyukYap.TabIndex = 14;
649:                    this.btnIlkHarfiBuyukYap.Text = "Ilk Harf Büyük";
650:                    this.btnIlkHarfiBuyukYap.UseVisualStyleBackColor = true;
651:                    this.btnIlkHarfiBuyukYap.Click += new
System.EventHandler(this.btnIlkHarfiBuyukYap_Click);
652:                    //
653:                    // label16
654:                    //
655:                    this.label16.AutoSize = true;
656:                    this.label16.Location = new System.Drawing.Point(24, 96);
657:                    this.label16.Margin = new System.Windows.Forms.Padding(4, 0, 4, 0);
658:                    this.label16.Name = "label16";
659:                    this.label16.Size = new System.Drawing.Size(48, 17);
660:                    this.label16.TabIndex = 13;
661:                    this.label16.Text = "Sonuç";
662:                    //
663:                    // lbMetinSonuc
664:                    //
665:                    this.lbMetinSonuc.FormattingEnabled = true;
666:                    this.lbMetinSonuc.HorizontalScrollbar = true;
667:                    this.lbMetinSonuc.ItemHeight = 16;
668:                    this.lbMetinSonuc.Location = new System.Drawing.Point(83, 96);
669:                    this.lbMetinSonuc.Margin = new System.Windows.Forms.Padding(4, 4, 4, 4);
670:                    this.lbMetinSonuc.Name = "lbMetinSonuc";
671:                    this.lbMetinSonuc.Size = new System.Drawing.Size(868, 100);
672:                    this.lbMetinSonuc.TabIndex = 12;
673:                    //
674:                    // btnIlkHarfleriBuyukYap
675:                    //
676:                    this.btnIlkHarfleriBuyukYap.Location = new System.Drawing.Point(28, 52);
677:                    this.btnIlkHarfleriBuyukYap.Margin = new System.Windows.Forms.Padding(4,
4, 4, 4);
678:                    this.btnIlkHarfleriBuyukYap.Name = "btnIlkHarfleriBuyukYap";
679:                    this.btnIlkHarfleriBuyukYap.Size = new System.Drawing.Size(139, 28);
680:                    this.btnIlkHarfleriBuyukYap.TabIndex = 11;
681:                    this.btnIlkHarfleriBuyukYap.Text = "Ilk Harfler Büyük";
682:                    this.btnIlkHarfleriBuyukYap.UseVisualStyleBackColor = true;
683:                    this.btnIlkHarfleriBuyukYap.Click += new
System.EventHandler(this.btnIlkHarfleriBuyukYap_Click);
684:                    //
685:                    // txtMetin
686:                    //
687:                    this.txtMetin.Location = new System.Drawing.Point(159, 15);
688:                    this.txtMetin.Margin = new System.Windows.Forms.Padding(4, 4, 4, 4);
689:                    this.txtMetin.Name = "txtMetin";
```

```
690:                    this.txtMetin.Size = new System.Drawing.Size(792, 22);
691:                    this.txtMetin.TabIndex = 4;
692:                    //
693:                    // label17
694:                    //
695:                    this.label17.AutoSize = true;
696:                    this.label17.Location = new System.Drawing.Point(23, 23);
697:                    this.label17.Margin = new System.Windows.Forms.Padding(4, 0, 4, 0);
698:                    this.label17.Name = "label17";
699:                    this.label17.Size = new System.Drawing.Size(42, 17);
700:                    this.label17.TabIndex = 3;
701:                    this.label17.Text = "Metin";
702:                    //
703:                    // btnEPostaKontrol
704:                    //
705:                    this.btnEPostaKontrol.Location = new System.Drawing.Point(858, 52);
706:                    this.btnEPostaKontrol.Margin = new System.Windows.Forms.Padding(4);
707:                    this.btnEPostaKontrol.Name = "btnEPostaKontrol";
708:                    this.btnEPostaKontrol.Size = new System.Drawing.Size(160, 28);
709:                    this.btnEPostaKontrol.TabIndex = 21;
710:                    this.btnEPostaKontrol.Text = "E-Posta Geçerli Mi?";
711:                    this.btnEPostaKontrol.UseVisualStyleBackColor = true;
712:                    this.btnEPostaKontrol.Click += new
System.EventHandler(this.btnEPostaKontrol_Click);
713:                    //
714:                    // FrmAnaSayfa
715:                    //
716:                    this.AutoScaleDimensions = new System.Drawing.SizeF(8F, 16F);
717:                    this.AutoScaleMode = System.Windows.Forms.AutoScaleMode.Font;
718:                    this.AutoSizeMode = System.Windows.Forms.AutoSizeMode.GrowAndShrink;
719:                    this.ClientSize = new System.Drawing.Size(1067, 844);
720:                    this.Controls.Add(this.pnlMetinIslemleri);
721:                    this.Controls.Add(this.pnlKisiIslemleri);
722:                    this.Controls.Add(this.pnlSayiIslemleri);
723:                    this.Controls.Add(this.pnlSifreIslemleri);
724:                    this.Controls.Add(this.pnlTarihIslemleri);
725:                    this.Controls.Add(this.label1);
726:                    this.Controls.Add(this.cmbIslemTuru);
727:                    this.Margin = new System.Windows.Forms.Padding(4, 4, 4, 4);
728:                    this.Name = "FrmAnaSayfa";
729:                    this.StartPosition = System.Windows.Forms.FormStartPosition.CenterScreen;
730:                    this.Text = "Islemler Ekrani";
731:                    this.Load += new System.EventHandler(this.FrmAnaSayfa_Load);
732:                    this.pnlTarihIslemleri.ResumeLayout(false);
733:                    this.pnlTarihIslemleri.PerformLayout();
734:                    this.pnlSifreIslemleri.ResumeLayout(false);
735:                    this.pnlSifreIslemleri.PerformLayout();
736:                    this.pnlSayiIslemleri.ResumeLayout(false);
737:                    this.pnlSayiIslemleri.PerformLayout();
738:                    this.pnlKisiIslemleri.ResumeLayout(false);
739:                    this.pnlKisiIslemleri.PerformLayout();
740:                    this.pnlMetinIslemleri.ResumeLayout(false);
741:                    this.pnlMetinIslemleri.PerformLayout();
742:                    this.ResumeLayout(false);
743:                    this.PerformLayout();
744:            }
745:
746:        private System.Windows.Forms.Button btnEPostaKontrol;
747:
748:        #endregion
749:
750:        private System.Windows.Forms.ComboBox cmbIslemTuru;
751:        private System.Windows.Forms.Label label1;
752:        private System.Windows.Forms.Panel pnlTarihIslemleri;
753:        private System.Windows.Forms.Label label2;
754:        private System.Windows.Forms.Label label3;
755:        private System.Windows.Forms.DateTimePicker dtpBaslangicTarihi;
756:        private System.Windows.Forms.DateTimePicker dtpBitisTarihi;
757:        private System.Windows.Forms.ListBox lbSonuc;
```

```
758:        private System.Windows.Forms.Label label4;
759:        private System.Windows.Forms.Button btnYasHesapla;
760:        private System.Windows.Forms.Button btnYasHesaplaMetinsel;
761:        private System.Windows.Forms.Button btnTarihAraligi;
762:        private System.Windows.Forms.Panel pnlSifreIslemleri;
763:        private System.Windows.Forms.Label label5;
764:        private System.Windows.Forms.TextBox txtSifrelenecekMetin;
765:        private System.Windows.Forms.Button btnMd5Hash;
766:        private System.Windows.Forms.Label label6;
767:        private System.Windows.Forms.ListBox lbSifreSonuc;
768:        private System.Windows.Forms.Button btnSha256Hash;
769:        private System.Windows.Forms.Button btnSha512Hash;
770:        private System.Windows.Forms.Button btnRastgeleSifreUret;
771:        private System.Windows.Forms.Button btnSifreUygunMu;
772:        private System.Windows.Forms.Label label8;
773:        private System.Windows.Forms.Label label7;
774:        private System.Windows.Forms.Button btnAyFarki;
775:        private System.Windows.Forms.Panel pnlSayiIslemleri;
776:        private System.Windows.Forms.Label label9;
777:        private System.Windows.Forms.Button btnOnayKoduOlustur;
778:        private System.Windows.Forms.Button btnMetneDonustur;
779:        private System.Windows.Forms.Label label10;
780:        private System.Windows.Forms.ListBox lbSayiSonuc;
781:        private System.Windows.Forms.Button btnSayisalMi;
782:        private System.Windows.Forms.TextBox txtSayi;
783:        private System.Windows.Forms.Label label11;
784:        private System.Windows.Forms.Panel pnlKisiIslemleri;
785:        private System.Windows.Forms.Label label12;
786:        private System.Windows.Forms.Button btnEPostaDogrula;
787:        private System.Windows.Forms.Button btnIbanDogrula;
788:        private System.Windows.Forms.Label label13;
789:        private System.Windows.Forms.ListBox lbKisiSonuc;
790:        private System.Windows.Forms.Button btnTcKimlikDogrula;
791:        private System.Windows.Forms.TextBox txtKisiMetni;
792:        private System.Windows.Forms.Label label14;
793:        private System.Windows.Forms.Panel pnlMetinIslemleri;
794:        private System.Windows.Forms.Label label15;
795:        private System.Windows.Forms.Button btnTurkceKarakterDuzelt;
796:        private System.Windows.Forms.Button btnIlkHarfiBuyukYap;
797:        private System.Windows.Forms.Label label16;
798:        private System.Windows.Forms.ListBox lbMetinSonuc;
799:        private System.Windows.Forms.Button btnIlkHarfleriBuyukYap;
800:        private System.Windows.Forms.TextBox txtMetin;
801:        private System.Windows.Forms.Label label17;
802:        private System.Windows.Forms.Button btnMetinSifrele;
803:        private System.Windows.Forms.Button btnTelefonNoDuzenle;
804:    }
805: }
806:
```

Namespaces

Name	Description
CSharpHelperMethods (⊞ see page 1)	This is namespace CSharpHelperMethods.

1.2.10 KisiIslemleri.cs

This is file KisiIslemleri.cs.

Body Source

```
1: ?using System;
2: using System.Text;
3: using System.Text.RegularExpressions;
4:
```

```
  5: namespace CSharpHelperMethods.Library
  6: {
  7:      /// <summary>
  8:      /// Kisi ile iliskili yardimci metotlarin tamamini içeren sinif
  9:      /// </summary>
 10:      public static class KisiIslemleri
 11:      {
 12:          /// <summary>
 13:          /// Girilen TC Kimlik Numarasinin, standart TC Kimlik No algoritmasina uygun olup
 14:          /// olmadigini kontrol eden metottur.
 15:          /// </summary>
 16:          /// <param name="tcKimlikNo">TC Kimlik No</param>
 17:          /// <returns>TC Kimlik No'nun Uygun Olup Olmadigi Bilgisi</returns>
 18:          public static bool DogrulaTcKimlikNo(string tcKimlikNo)
 19:          {
 20:              var tcNo = Convert.ToInt64(tcKimlikNo);
 21:              var atcno = tcNo / 100;
 22:              var btcno = tcNo / 100;
 23:              var c1 = atcno % 10; atcno /= 10;
 24:              var c2 = atcno % 10; atcno /= 10;
 25:              var c3 = atcno % 10; atcno /= 10;
 26:              var c4 = atcno % 10; atcno /= 10;
 27:              var c5 = atcno % 10; atcno /= 10;
 28:              var c6 = atcno % 10; atcno /= 10;
 29:              var c7 = atcno % 10; atcno /= 10;
 30:              var c8 = atcno % 10; atcno /= 10;
 31:              var c9 = atcno % 10;
 32:              var q1 = (10 - ((c1 + c3 + c5 + c7 + c9) * 3 + c2 + c4 + c6 + c8) % 10) % 10;
 33:              var q2 = (10 - ((c2 + c4 + c6 + c8 + q1) * 3 + c1 + c3 + c5 + c7 + c9) % 10) % 10;
 34:              var dogrula = btcno * 100 + q1 * 10 + q2 == tcNo;
 35:              return dogrula;
 36:          }
 37:
 38:          /// <summary>
 39:          /// Girilen IBAN'in standar IBAN algoritmasina uygun olup olmadigini dogrular
 40:          /// </summary>
 41:          /// <code>DogrulaIBAN("TR560006200000012990022604")</code>
 42:          /// <param name="iban">Uluslar arasi banka hesap numarasi (IBAN) </param>
 43:          /// <returns>IBAN uygunsa true, degilse false degeri döner</returns>
 44:          public static bool DogrulaIBAN(string iban)
 45:          {
 46:              iban = iban.ToUpper();
 47:
 48:              if (string.IsNullOrEmpty(iban))
 49:                  return false;
 50:
 51:              if (!Regex.IsMatch(iban, "^[A-Z0-9]"))
 52:                  return false;
 53:
 54:              iban = iban.Replace(" ", String.Empty);
 55:              var bank = iban.Substring(4, iban.Length - 4) + iban.Substring(0, 4);
 56:              const int asciiShift = 55;
 57:              var sb = new StringBuilder();
 58:              foreach (var c in bank)
 59:              {
 60:                  sb.Append(char.IsLetter(c) ? c - asciiShift : int.Parse(c.ToString()));
 61:              }
 62:
 63:              var checkSumString = sb.ToString();
 64:              var checksum = int.Parse(checkSumString.Substring(0, 1));
 65:              for (var i = 1; i < checkSumString.Length; i++)
 66:              {
 67:                  var v = int.Parse(checkSumString.Substring(i, 1));
 68:                  checksum *= 10;
 69:                  checksum += v;
 70:                  checksum %= 97;
```

```
71:                }
72:                return checksum == 1;
73:            }
74:
75:            /// <summary>
76:            /// Ekranlardan girilen bir e-posta adresinin geçerli bir e-posta adresi olup
olmadigini
77:            /// Kontrol eden metottur
78:            /// </summary>
79:            /// <param name="ePosta">E-Posta Adresi</param>
80:            /// <returns>E-Posta Adresinin Geçerli Olup Olmadigi Bilgisi</returns>
81:            public static bool DogrulaEPosta(string ePosta)
82:            {
83:                const string pattern = @"^(([\w-]+\.)+[\w-]+|([a-zA-Z]{1}|[\w-]{2,}))@"
84:                                       + @"((([0-1]?[0-9]{1,2}|25[0-5]|2[0-4][0-9])\."
85:                                       + @"([0-1]?[0-9]{1,2}|25[0-5]|2[0-4][0-9])\."
86:                                       + @"([0-1]?[0-9]{1,2}|25[0-5]|2[0-4][0-9])\."
87:                                       + @"([0-1]?[0-9]{1,2}|25[0-5]|2[0-4][0-9])){1}|"
88:                                       + @"([a-zA-Z]+[\w-]+\.)+[a-zA-Z]{2,4})$";
89:                return Regex.IsMatch(ePosta, pattern);
90:            }
91:        }
92: }
```

Namespaces

Name	Description
CSharpHelperMethods (🖻 see page 1)	This is namespace CSharpHelperMethods.

1.2.11 **MetinIslemleri.cs**

This is file MetinIslemleri.cs.

Body Source

```
1: ?using System;
2: using System.Collections.Generic;
3: using System.Globalization;
4: using System.Linq;
5: using System.Net.Mail;
6: using System.Text;
7: using System.Text.RegularExpressions;
8:
9: namespace CSharpHelperMethods.Library
10: {
11:     /// <summary>
12:     /// Metinsel veri kontrol metotlarini barindiran sinif
13:     /// </summary>
14:     public static class MetinIslemleri
15:     {
16:         /// <summary>
17:         /// Girilen metnin basinda virgül olup olmadigini kontrol eden metottur.
18:         /// </summary>
19:         /// <param name="text">Girdi metni</param>
20:         /// <returns>Girilen Metnin Olmasi Gerektigi Hali</returns>
21:         public static string IdealMetinGetir(string text)
22:         {
23:             if (!string.IsNullOrEmpty(text) && text[0] == ',')
24:             {
25:                 text = text.Substring(1, text.Length - 1);
26:             }
27:             return text;
28:         }
29:
30:         public static string MetinZararliKarakterDuzelt(string text)
```

```
31:                  {
32:                      text = text.Replace("&#252;", "ü");
33:                      return text;
34:                  }
35:
36:              public static string MetinSifrele(string metin)
37:                  {
38:                      if (string.IsNullOrEmpty(metin)) return metin;
39:                      var metinDuzenleyici = new StringBuilder(metin);
40:                      switch (metin.Length)
41:                      {
42:                          case 10:
43:                              metinDuzenleyici.Remove(2, 6);
44:                              metinDuzenleyici.Insert(2, "******");
45:                              metin = metinDuzenleyici.ToString();
46:                              break;
47:                          case 11:
48:                              metinDuzenleyici.Remove(2, 7);
49:                              metinDuzenleyici.Insert(2, "*******");
50:                              metin = metinDuzenleyici.ToString();
51:                              break;
52:                      }
53:                      return metin;
54:                  }
55:
56:              /// <summary>
57:              /// Girilen metnin basinda virgül olup olmadigini kontrol eden metottur.
58:              /// </summary>
59:              /// <param name="text">Girdi metni</param>
60:              /// <returns>Girilen Metnin Olmasi Gerektigi Hali</returns>
61:              public static string TurkceKarakterleriDuzelt(string text)
62:                  {
63:                      text = text.Replace("ü", "u");
64:                      text = text.Replace("i", "i");
65:                      text = text.Replace("ö", "o");
66:                      text = text.Replace("ü", "u");
67:                      text = text.Replace("s", "s");
68:                      text = text.Replace("g", "g");
69:                      text = text.Replace("ç", "c");
70:                      text = text.Replace("Ü", "U");
71:                      text = text.Replace("I", "I");
72:                      text = text.Replace("Ö", "O");
73:                      text = text.Replace("Ü", "U");
74:                      text = text.Replace("S", "S");
75:                      text = text.Replace("G", "G");
76:                      text = text.Replace("Ç", "C");
77:                      return text;
78:                  }
79:
80:              /// <summary>
81:              /// Girilen metnin basinda virgül olup olmadigini kontrol eden metottur.
82:              /// </summary>
83:              /// <param name="text">Girdi metni</param>
84:              /// <returns>Girilen Metnin Olmasi Gerektigi Hali</returns>
85:              public static string UTF8Duzelt(string text)
86:                  {
87:                      text = text.Replace("&amp;ccedil;", "ç");
88:                      text = text.Replace("&amp;ouml;", "ö");
89:                      text = text.Replace("&amp;uuml;", "ü");
90:                      text = text.Replace("&ccedil;", "ç");
91:                      text = text.Replace("&ouml;", "ö");
92:                      text = text.Replace("&uuml;", "ü");
93:                      return text;
94:                  }
95:
96:              /// <summary>
97:              /// Girilen metnin basinda virgül olup olmadigini kontrol eden metottur.
98:              /// </summary>
99:              /// <param name="text">Girdi metni</param>
```

```
100:          /// <returns>Girilen Metnin Olmasi Gerektigi Hali</returns>
101:          public static string HTMLDuzelt(string text)
102:          {
103:              var sc = new List<string>
104:                  {
105:                      // get rid of unnecessary tag spans (comments and title)
106:                      @"<!--(\w|\W)+?-->",
107:                      @"<title>(\w|\W)+?</title>",
108:                      // Get rid of classes and styles
109:                      @"\s?class=\w+",
110:                      @"\s+style='[^']+'",
111:                      // Get rid of unnecessary tags
112:
@"<(meta|link|/?o:|/?style|/?div|/?st\d|/?head|/?html|body|/?body|/?span|!\[)[^>]*?>",
113:                      // Get rid of empty paragraph tags
114:                      @"(<[^>]+>)+ (</\w+>)+",
115:                      @"<a[^>]*>([^<]+)</a>",
116:                      @"<p[^>]*>([^<]+)</p>",
117:                      // remove bizarre v: element attached to <img> tag
118:                      @"\s+v:\w+=""[^""]+""",
119:                      // remove extra lines
120:                      @"(\n\r){2,}",
121:                      " "
122:                  };
123:              text = sc.Aggregate(text, (current, s) => Regex.Replace(current, s,
string.Empty, RegexOptions.IgnoreCase));
124:              text = Regex.Replace(text, "<.*?>", String.Empty);
125:              text = text.Replace("\r", " ").Replace("\n", " ").Replace("  ", "
").Trim();
126:              text = text.Replace("&rsquo;", "'");
127:              text = text.Replace("rsquo;", "'");
128:              text = text.Replace("&rdquo;", "\"");
129:              text = text.Replace("rdquo;", "\"");
130:
131:              text = text.Replace("&lsquo;", "'");
132:              text = text.Replace("lsquo;", "'");
133:              text = text.Replace("&ldquo;", "\"");
134:              text = text.Replace("ldquo;", "\"");
135:
136:              text = text.Replace("&lt;", "<");
137:              text = text.Replace("&gt;", ">");
138:              text = text.Replace("&amp;", "&");
139:              //text = text.Replace("&lt;", "");
140:              //text = text.Replace("p&gt;", "");
141:              //text = text.Replace("/p&gt;", "");
142:              //text = text.Replace("&gt;", "");
143:              //return text;
144:              return text;
145:          }
146:
147:          public static string MetinIlkKarakterBuyukDigerleriKucukYap(string text) =>
text.First().ToString().ToUpper() + text.Substring(1);
148:
149:          /// <summary>
150:          /// Girilen bir metindeki tüm kelimelerin ilk harflerinin
151:          /// Büyük yapilmasi için hazirlanan metottur
152:          /// </summary>
153:          /// <param name="input">Girilen metin</param>
154:          /// <returns>Metnin Kelimelerinin Ilk Harflerinin Büyültülmüs Hali</returns>
155:          public static string MetinIlkKarakterleriBuyukYap(this string input)
156:          {
157:              var metinDuzenleyici = new StringBuilder();
158:              var wordList = input.Split(' ');
159:              var cultureInfo = new CultureInfo("tr-TR");
160:              foreach (var item in wordList)
161:              {
162:                  if (item.Length > 1)
163:                  {
164:                      metinDuzenleyici.Append(item.Substring(0, 1).ToUpper(cultureInfo));
```

```
165:                         metinDuzenleyici.Append(item.Substring(1).ToLower(cultureInfo));
166:                         metinDuzenleyici.Append(" ");
167:                     }
168:                     else if (item.Length == 1)
169:                     {
170:                         metinDuzenleyici.Append(item.ToUpper(cultureInfo));
171:                         metinDuzenleyici.Append(" ");
172:                     }
173:                 }
174:             return metinDuzenleyici.ToString();
175:         }
176:
177:         /// <summary>
178:         /// Telefon No, E-Posta Adresi vb. metinsel veriler içerisindeki
179:         /// Bosluk, parantez, kisa çizgi gibi verilerin silinmesi için hazirlanmis
metottur
180:         /// </summary>
181:         /// <param name="metin">Girdi Metin Bilgisi</param>
182:         /// <returns>Çikti Metin Bilgisi</returns>
183:         private static string GereksizKarakterTemizle(string metin)
184:         {
185:             return metin.Replace("-", "").Replace(" ", "").Replace("(",
"").Replace(")", "");
186:         }
187:
188:         /// <summary>
189:         /// Girilen telefon numarasinin SMS gönderilecek hale getirilmesini saglayan
metottur.
190:         /// </summary>
191:         /// <param name="telefonNo">Telefon No Bilgisi</param>
192:         /// <returns>Girilen Telefon Numarasinin SMS Gönderilecek Hali</returns>
193:         public static string TelefonNoDuzenle(string telefonNo)
194:         {
195:             if (string.IsNullOrEmpty(telefonNo)) return string.Empty;
196:             telefonNo = GereksizKarakterTemizle(telefonNo);
197:             if (telefonNo.Length == 11)
198:             {
199:                 telefonNo = telefonNo.Substring(telefonNo.Length - 10);
200:             }
201:             return telefonNo;
202:
203:         }
204:
205:         /// <summary>
206:         /// Girilen metnin bosluklarinin kaldirilmasini saglayan metottur.
207:         /// </summary>
208:         /// <param name="metin">Telefon No Bilgisi</param>
209:         /// <returns>Girilen Metnin Istenen Hali</returns>
210:         public static string BoslukKaldir(string metin)
211:         {
212:             if (string.IsNullOrEmpty(metin)) return string.Empty;
213:             metin = GereksizKarakterTemizle(metin);
214:             return metin;
215:         }
216:
217:         /// <summary>
218:         /// Girilen mail adresinin geçerli olup olmadigini kontrol eden bir metottur
219:         /// </summary>
220:         /// <param name="email">E-Posta Adresi</param>
221:         /// <returns>Geçerli veya Degil Bilgisi</returns>
222:         public static bool MailAdresiGecerliMi(string email)
223:         {
224:             var gecerliMi = true;
225:             try
226:             {
227:                 var emailAddress = new MailAddress(email);
228:             }
229:             catch
230:             {
```

```
231:                    gecerliMi = false;
232:                }
233:            return gecerliMi;
234:        }
235:    }
236: }
```

Namespaces

Name	Description
CSharpHelperMethods (⊠ see page 1)	This is namespace CSharpHelperMethods.

1.2.12 MetinIslemleriTest.cs

This is file MetinIslemleriTest.cs.

Body Source

```
 1: ?using CSharpHelperMethods.Library;
 2: using NUnit.Framework;
 3:
 4: namespace CSharpHelperMethods.Test
 5: {
 6:     /// <summary>
 7:     /// Library kisminda Metin Islemleri bölümünde
 8:     /// Yer alan metotlarin örnek test durumlari ve beklenen sonuçlari ile
 9:     /// Test edildikleri siniftir
10:     /// </summary>
11:     [TestFixture]
12:     public class MetinIslemleriTest
13:     {
14:         [Test]
15:         [TestCase("serdargul@outlook.com", ExpectedResult = true)]
16:         [TestCase("serdar@aol.com", ExpectedResult = true)]
17:         [TestCase("a.b", ExpectedResult = false)]
18:         [TestCase("a_c", ExpectedResult = false)]
19:         public bool MailAdresiGecerliMiTest(string metin)
20:         {
21:             return MetinIslemleri.MailAdresiGecerliMi(metin);
22:         }
23:
24:         [Test]
25:         [TestCase("2013123817", ExpectedResult = "20******17")]
26:         [TestCase("11111111111", ExpectedResult = "11*******11")]
27:         [TestCase("serdar", ExpectedResult = "serdar")]
28:         public string MetinSifreleTest(string metin)
29:         {
30:             return MetinIslemleri.MetinSifrele(metin);
31:         }
32:
33:         [Test]
34:         [TestCase("2013123817  ", ExpectedResult = "2013123817")]
35:         [TestCase("ses ss", ExpectedResult = "sesss")]
36:         public string BoslukKaldirTest(string metin)
37:         {
38:             return MetinIslemleri.BoslukKaldir(metin);
39:         }
40:
41:         [Test]
42:         [TestCase("2013123817", ExpectedResult = "2013123817")]
43:         [TestCase("sesss&#252;", ExpectedResult = "sesssü")]
44:         public string MetinZararliKarakterDuzeltTest(string metin)
45:         {
46:             return MetinIslemleri.MetinZararliKarakterDuzelt(metin);
47:         }
```

```
48:
49:            [Test]
50:            [TestCase("serdar", ExpectedResult = "Serdar")]
51:            [TestCase("bu BIR test metnidir", ExpectedResult = "Bu BIR test metnidir")]
52:            public string MetinIlkKarakterBuyukDigerleriKucukYapTest(string metin)
53:            {
54:                return MetinIslemleri.MetinIlkKarakterBuyukDigerleriKucukYap(metin);
55:            }
56:
57:            [Test]
58:            [TestCase("serdar", ExpectedResult = "Serdar ")]
59:            [TestCase("bu BIR test metnidir", ExpectedResult = "Bu Bir Test Metnidir ")]
60:            public string MetinIlkKarakterleriBuyukYapTest(string metin)
61:            {
62:                return MetinIslemleri.MetinIlkKarakterleriBuyukYap(metin);
63:            }
64:
65:            [Test]
66:            [TestCase("serdar", ExpectedResult = "serdar")]
67:            [TestCase(",test metni", ExpectedResult = "test metni")]
68:            public string IdealMetinGetirTest(string metin)
69:            {
70:                return MetinIslemleri.IdealMetinGetir(metin);
71:            }
72:
73:            [Test]
74:            [TestCase("0(505)333-44-55", ExpectedResult = "5053334455")]
75:            [TestCase("0 505 123 45 67", ExpectedResult = "5051234567")]
76:            [TestCase("0-312-123-45-67", ExpectedResult = "3121234567")]
77:            public string TelefonNoDuzenleTest(string metin)
78:            {
79:                return MetinIslemleri.TelefonNoDuzenle(metin);
80:            }
81:
82:            [Test]
83:            [TestCase("fistikçi sahap", ExpectedResult = "fistikci sahap")]
84:            [TestCase("ugulu", ExpectedResult = "ugulu")]
85:            [TestCase("ögrenci", ExpectedResult = "ogrenci")]
86:            public string TurkceKarakterleriDuzeltTest(string metin)
87:            {
88:                return MetinIslemleri.TurkceKarakterleriDuzelt(metin);
89:            }
90:        }
91: }
```

Namespaces

Name	Description
CSharpHelperMethods (⊡ see page 1)	This is namespace CSharpHelperMethods.

1.2.13 Program.cs

This is file Program.cs.

Body Source

```
1: ?using System;
2: using System.Windows.Forms;
3:
4: namespace CSharpHelperMethods.UI
5: {
6:     internal static class Program
7:     {
8:         /// <summary>
9:         /// The main entry point for the application.
```

```
10:          /// </summary>
11:          [STAThread]
12:          private static void Main()
13:          {
14:              Application.EnableVisualStyles();
15:              Application.SetCompatibleTextRenderingDefault(false);
16:              Application.Run(new FrmAnaSayfa());
17:          }
18:      }
19: }
```

Namespaces

Name	Description
CSharpHelperMethods (▣ see page 1)	This is namespace CSharpHelperMethods.

1.2.14 **Resources.Designer.cs**

This code was generated by a tool. Runtime Version:4.0.30319.42000

Changes to this file may cause incorrect behavior and will be lost if the code is regenerated.

Body Source

```
 1: ?//------------------------------------------------------------------------------
 2: // <auto-generated>
 3: //     This code was generated by a tool.
 4: //     Runtime Version:4.0.30319.42000
 5: //
 6: //     Changes to this file may cause incorrect behavior and will be lost if
 7: //     the code is regenerated.
 8: // </auto-generated>
 9: //------------------------------------------------------------------------------
10:
11: namespace CSharpHelperMethods.Properties
12: {
13:
14:
15:     /// <summary>
16:     ///   A strongly-typed resource class, for looking up localized strings, etc.
17:     /// </summary>
18:     // This class was auto-generated by the StronglyTypedResourceBuilder
19:     // class via a tool like ResGen or Visual Studio.
20:     // To add or remove a member, edit your .ResX file then rerun ResGen
21:     // with the /str option, or rebuild your VS project.
22:
[global::System.CodeDom.Compiler.GeneratedCodeAttribute("System.Resources.Tools.StronglyType
dResourceBuilder",
"4.0.0.0")]
23:     [global::System.Diagnostics.DebuggerNonUserCodeAttribute()]
24:     [global::System.Runtime.CompilerServices.CompilerGeneratedAttribute()]
25:     internal class Resources
26:     {
27:
28:         private static global::System.Resources.ResourceManager resourceMan;
29:
30:         private static global::System.Globalization.CultureInfo resourceCulture;
31:
32:
[global::System.Diagnostics.CodeAnalysis.SuppressMessageAttribute("Microsoft.Performance",
"CA1811:AvoidUncalledPrivateCode")]
33:         internal Resources()
34:         {
35:         }
36:
```

```
37:            /// <summary>
38:            ///    Returns the cached ResourceManager instance used by this class.
39:            /// </summary>
40:
[global::System.ComponentModel.EditorBrowsableAttribute(global::System.ComponentModel.Editor
BrowsableState.Advanced)]
41:            internal static global::System.Resources.ResourceManager ResourceManager
42:            {
43:                get
44:                {
45:                    if ((resourceMan == null))
46:                    {
47:                        global::System.Resources.ResourceManager temp = new
global::System.Resources.ResourceManager("CSharpHelperMethods.UI.Properties.Resources",
typeof(Resources).Assembly);
48:                        resourceMan = temp;
49:                    }
50:                    return resourceMan;
51:                }
52:            }
53:
54:            /// <summary>
55:            ///    Overrides the current thread's CurrentUICulture property for all
56:            ///    resource lookups using this strongly typed resource class.
57:            /// </summary>
58:
[global::System.ComponentModel.EditorBrowsableAttribute(global::System.ComponentModel.Editor
BrowsableState.Advanced)]
59:            internal static global::System.Globalization.CultureInfo Culture
60:            {
61:                get
62:                {
63:                    return resourceCulture;
64:                }
65:                set
66:                {
67:                    resourceCulture = value;
68:                }
69:            }
70:        }
71: }
```

Namespaces

Name	Description
CSharpHelperMethods (⊡ see page 1)	This is namespace CSharpHelperMethods.

1.2.15 Sabitler.cs

This is file Sabitler.cs.

Body Source

```
1: ?namespace CSharpHelperMethods.Library
2: {
3:     /// <summary>
4:     /// Metin Sabitlerinin Bulundugu Sinif
5:     /// </summary>
6:     public static class Sabitler
7:     {
8:         public const string BaslangicTarihiGirilmedi = "Baslangiç Tarihi girilmedigi
için bu islem gerçeklestirilemez.";
9:         //public const string BitisTarihiGirilmedi = "Bitis Tarihi girilmedigi için bu
islem gerçeklestirilemez.";
10:         public const string BaslangicVeyaBitisTarihiGirilmedi = "Baslangiç Tarihi veya
```

```
      Bitis Tarihi girilmedigi için bu islem gerçeklestirilemez.";
11:         public const string BaslangicTarihiBuyuk = "Baslangiç Tarihi Bitis Tarihinden
      büyük olamaz.";
12:         public const string SifrelenecekMetinGirilmedi = "Sifrelenecek metin
      girilmedigi için bu islem gerçeklestirilemez.";
13:         public const string SifreMetniGirilmedi = "Sifre metni girilmedigi için bu
      islem gerçeklestirilemez.";
14:         public const string MetinGirilmedi = "Metin girilmedigi için bu islem
      gerçeklestirilemez.";
15:         public const string MetinSayisalDegil = "Girilen metin sayisal bir deger
      olmadigi için bu islem gerçeklestirilemez.";
16:         public const string MetinTcKimlikNoDegil = "Girilen metin 11 karakter olmadigi
      için bu islem gerçeklestirilemez. Karakter Sayisi: ";
17:         public const string MetinIbanNoDegil = "Girilen metin 26 karakter olmadigi için
      bu islem gerçeklestirilemez.";
18:         public const string MetinTcKimlikNoVeyaVergiNoDegil = "Girilen metin 10 veya 11
      karakter olmadigi için bu islem gerçeklestirilemez.";
19:         public const string EPostaAdresiGecerli = "Girilen e-posta adresi geçerli bir
      e-posta adresidir";
20:         public const string EPostaAdresiGecerliDegil = "Girilen e-posta adresi geçerli
      bir e-posta adresi degildir";
21:     }
22: }
```

Namespaces

Name	Description
CSharpHelperMethods (⊠ see page 1)	This is namespace CSharpHelperMethods.

1.2.16 SayiIslemleri.cs

This is file SayiIslemleri.cs.

Body Source

```
 1: ?using System;
 2: using System.Globalization;
 3: using System.Linq;
 4: using System.Text;
 5:
 6: namespace CSharpHelperMethods.Library
 7: {
 8:     /// <summary>
 9:     /// Sayisal veri kontrol metotlarini barindiran sinif
10:     /// </summary>
11:     public static class SayiIslemleri
12:     {
13:         /// <summary>
14:         /// Girilen bir metnin sayi olup olmadigini kontrol eden metottur.
15:         /// </summary>
16:         /// <param name="text">Girdi metni</param>
17:         /// <returns>Girilen Metnin Sayisal Olup Olmadigi Bilgisi</returns>
18:         public static bool SayisalMi(string text) => text.All(char.IsNumber);
19:
20:         /// <summary>
21:         /// Ekrandan alinan string degerde virgül oldugundan ajax metodu patladigi için
22:         /// Burada string olarak tutar verisi aliniyor
23:         /// Ve decimala çevriliyor
24:         /// </summary>
25:         /// <param name="metin">Girilen String Deger</param>
26:         /// <returns>Decimal Deger</returns>
27:         public static decimal SayiHalineCevir(string metin)
28:         {
29:             decimal val;
30:             if (metin.Contains("_") || metin.Contains(".") || metin.Contains("."))
```

```
 31:                    {
 32:                        if (!decimal.TryParse(metin.Replace(",", "").Replace(".",
""").Replace("_", ""), NumberStyles.Number,
 33:                            CultureInfo.InvariantCulture, out val))
 34:                            return 0;
 35:                        return val / 100;
 36:                    }
 37:
 38:                    var value = decimal.TryParse(metin.Replace(",", "").Replace(".",
""").Replace("_", ""), NumberStyles.Number,
 39:                        CultureInfo.InvariantCulture, out val);
 40:                    return value ? val : 0;
 41:                }
 42:
 43:            /// <summary>
 44:            /// Elimizdeki tutar verilerinin
 45:            /// Yazi formatinda gösterilmesini saglayacak bir metottur
 46:            /// </summary>
 47:            /// <param name="tutar">Girilen Sayi Bilgisi</param>
 48:            /// <returns>Sayinin Yaziya Dönüstürülmüs Hali</returns>
 49:            public static string SayiyiYaziyaCevir(decimal tutar)
 50:                {
 51:                    var sTutar = tutar.ToString("F2").Replace('.', ','); // Replace('.',',')
ondalik ayracinin . olma durumu için
 52:                    var lira = sTutar.Substring(0, sTutar.IndexOf(',')); //tutarin tam kismi
 53:                    var kurus = sTutar.Substring(sTutar.IndexOf(',') + 1, 2);
 54:                    var yazi = new StringBuilder();
 55:
 56:                    var birler = new[] {"", "BIR", "IKI", "ÜÇ", "DÖRT", "BES", "ALTI", "YEDI",
"SEKIZ", "DOKUZ"};
 57:                    var onlar = new[] {"", "ON", "YIRMI", "OTUZ", "KIRK", "ELLI", "ALTMIS",
"YETMIS", "SEKSEN", "DOKSAN"};
 58:                    var binler = new[] {"KENTRILYON", "KATRILYON", "TRILYON", "MILYAR",
"MILYON", "BIN", ""};
 59:
 60:                    const int grupSayisi = 7;
 61:                    //sayidaki 3'lü grup sayisi. katrilyon içi 6. (1.234,00 daki grup sayisi
2'dir.)
 62:                    //KATRILYON'un basina ekleyeceginiz her deger için grup sayisini artiriniz.
 63:
 64:                    lira = lira.PadLeft(grupSayisi * 3, '0'); //sayinin soluna '0' eklenerek
sayi 'grup sayisi x 3' basakmakli yapiliyor.
 65:
 66:                    for (var i = 0; i < grupSayisi * 3; i += 3) //sayi 3'erli gruplar halinde
ele aliniyor.
 67:                    {
 68:                        var grupDegeri = new StringBuilder();
 69:
 70:                        if (lira.Substring(i, 1) != "0")
 71:                            grupDegeri.Append(birler[Convert.ToInt32(lira.Substring(i, 1))] +
"YÜZ"); //yüzler
 72:                        if (grupDegeri.ToString().Equals("BIRYÜZ"))
 73:                            grupDegeri.Replace("BIRYÜZ", "YÜZ");
 74:                        grupDegeri.Append(onlar[Convert.ToInt32(lira.Substring(i + 1, 1))]);
//onlar
 75:                        grupDegeri.Append(birler[Convert.ToInt32(lira.Substring(i + 2,
1))]);//birler
 76:                        if (grupDegeri.Length > 0) //binler
 77:                            grupDegeri.Append(binler[i / 3]);
 78:                        if (grupDegeri.ToString().Equals("BIRBIN"))
 79:                            grupDegeri.Replace("BIRBIN", "BIN");
 80:
 81:                        yazi.Append(grupDegeri);
 82:                    }
 83:
 84:                    if (yazi.Length > 0)
 85:                        yazi.Append(" TL ");
 86:
 87:                    var yaziUzunlugu = yazi.Length;
```

1

```
 88:
 89:                if (kurus.Substring(0, 1) != "0") //kurus onlar
 90:                    yazi.Append(onlar[Convert.ToInt32(kurus.Substring(0, 1))]);
 91:
 92:                if (kurus.Substring(1, 1) != "0") //kurus birler
 93:                    yazi.Append(birler[Convert.ToInt32(kurus.Substring(1, 1))]);
 94:
 95:                yazi.Append(yazi.Length > yaziUzunlugu ? " KR." : "SIFIR KR.");
 96:
 97:                return yazi.ToString();
 98:            }
 99:
100:            /// <summary>
101:            /// Sadece rakamlardan olusan 6 haneli random sayi üreten metottur.
102:            /// </summary>
103:            /// <returns>6 Haneli Onay Kodu</returns>
104:            public static string AltiHaneliOnayKoduOlustur()
105:            {
106:                var rastgele = new Random();
107:                var sb = new StringBuilder();
108:                for (var i = 0; i < 6; i++)
109:                {
110:                    var ascii = rastgele.Next(48, 57); //Rakamlar
111:                    var karakter = Convert.ToChar(ascii);
112:                    sb.Append(karakter);
113:                }
114:
115:                return sb.ToString();
116:            }
117:        }
118: }
```

Namespaces

Name	Description
CSharpHelperMethods (⊡ see page 1)	This is namespace CSharpHelperMethods.

1.2.17 **SayiIslemleriTest.cs**

This is file SayiIslemleriTest.cs.

Body Source

```
 1: ?using CSharpHelperMethods.Library;
 2: using NUnit.Framework;
 3:
 4: namespace CSharpHelperMethods.Test
 5: {
 6:     /// <summary>
 7:     /// Library kisminda Sayi Islemleri bölümünde
 8:     /// Yer alan metotlarin örnek test durumlari ve beklenen sonuçlari ile
 9:     /// Test edildikleri siniftir
10:     /// </summary>
11:     [TestFixture]
12:     public class SayiIslemleriTest
13:     {
14:         [Test]
15:         [TestCase("15", ExpectedResult = true)]
16:         [TestCase("15123132", ExpectedResult = true)]
17:         [TestCase("3242a", ExpectedResult = false)]
18:         [TestCase("asdada", ExpectedResult = false)]
19:         public bool SayisalMiTest(string metin)
20:         {
21:             return SayiIslemleri.SayisalMi(metin);
22:         }
```

```
23:
24:          [Test]
25:          [TestCase("15_05", ExpectedResult = 15.05)]
26:          [TestCase("456.11", ExpectedResult = 456.11)]
27:          public decimal SayiHalineCevirTest(string metin)
28:          {
29:              return SayiIslemleri.SayiHalineCevir(metin);
30:          }
31:
32:          [Test]
33:          [TestCase(1500, ExpectedResult = "BINBESYÜZ TL SIFIR KR.")]
34:          [TestCase(456789.76, ExpectedResult = "DÖRTYÜZELLIALTIBINYEDIYÜZSEKSENDOKUZ TL
YETMISALTI KR.")]
35:          public string SayiyiYaziyaCevirTest(decimal sayi)
36:          {
37:              return SayiIslemleri.SayiyiYaziyaCevir(sayi);
38:          }
39:      }
40: }
```

Namespaces

Name	Description
CSharpHelperMethods (⊡ see page 1)	This is namespace CSharpHelperMethods.

1.2.18 Settings.Designer.cs

This code was generated by a tool. Runtime Version:4.0.30319.42000

Changes to this file may cause incorrect behavior and will be lost if the code is regenerated.

Body Source

```
 1: ?//------------------------------------------------------------------------------
 2: // <auto-generated>
 3: //     This code was generated by a tool.
 4: //     Runtime Version:4.0.30319.42000
 5: //
 6: //     Changes to this file may cause incorrect behavior and will be lost if
 7: //     the code is regenerated.
 8: // </auto-generated>
 9: //------------------------------------------------------------------------------
10:
11: namespace CSharpHelperMethods.Properties
12: {
13:
14:
15:     [global::System.Runtime.CompilerServices.CompilerGeneratedAttribute()]
16:
[global::System.CodeDom.Compiler.GeneratedCodeAttribute("Microsoft.VisualStudio.Editors.Sett
ingsDesigner.SettingsSingleFileGenerator",
"11.0.0.0")]
17:     internal sealed partial class Settings :
global::System.Configuration.ApplicationSettingsBase
18:     {
19:
20:         private static Settings defaultInstance =
((Settings)(global::System.Configuration.ApplicationSettingsBase.Synchronized(new
Settings())));
21:
22:         public static Settings Default
23:         {
24:             get
25:             {
26:                 return defaultInstance;
```

```
27:                    }
28:                }
29:            }
30: }
```

Namespaces

Name	Description
CSharpHelperMethods (⊡ see page 1)	This is namespace CSharpHelperMethods.

1.2.19 SifrelemeIslemleri.cs

This is file SifrelemeIslemleri.cs.

Body Source

```
 1: ?using System;
 2: using System.Collections.Generic;
 3: using System.Linq;
 4: using System.Security.Cryptography;
 5: using System.Text;
 6: using System.Text.RegularExpressions;
 7:
 8: namespace CSharpHelperMethods.Library
 9: {
10:     /// <summary>
11:     /// Sifre ile ilgili islemler sinifi
12:     /// </summary>
13:     public static class SifrelemeIslemleri
14:     {
15:         /// <summary>
16:         /// Kullanici sifresinin SHA yöntemiyle sifrelenmesini saglayan metottur.
17:         /// </summary>
18:         /// <param name="text">Sifrelenecek Metin</param>
19:         /// <returns>Girdi Metninin Sifrelenmis Hali</returns>
20:         public static string HesaplaSHA512(string text)
21:         {
22:             var sha512 = new SHA512Managed();
23:             var encryptedSha512 = sha512.ComputeHash(Encoding.UTF8.GetBytes(text));
24:             var strHex = encryptedSha512.Aggregate("", (current, b) =>
$"{current}{b:x2}");
25:             return strHex.ToUpper();
26:         }
27:
28:         /// <summary>
29:         /// Kullanici sifresinin SHA yöntemiyle sifrelenmesini saglayan metottur.
30:         /// </summary>
31:         /// <param name="text">Sifrelenecek Metin</param>
32:         /// <returns>Girdi Metninin Sifrelenmis Hali</returns>
33:         public static string HesaplaSHA256(string text)
34:         {
35:             var sha256 = SHA256.Create();
36:             var bytes = Encoding.UTF8.GetBytes(text);
37:             var hash = sha256.ComputeHash(bytes);
38:             var result = new StringBuilder();
39:             foreach (var t in hash)
40:             {
41:                 result.Append(t.ToString("X2"));
42:             }
43:             return result.ToString();
44:         }
45:
46:         /// <summary>
47:         /// Kullanici sifresinin MD5 yöntemiyle sifrelenmesini saglayan metottur.
48:         /// </summary>
```

```
 49:          /// <param name="text">Sifrelenecek Metin</param>
 50:          /// <returns>Girdi Metninin Sifrelenmis Hali</returns>
 51:          public static string HesaplaMD5(string text)
 52:          {
 53:              var md5 = MD5.Create();
 54:              var inputBytes = Encoding.ASCII.GetBytes(text);
 55:              var hash = md5.ComputeHash(inputBytes);
 56:
 57:              var sb = new StringBuilder();
 58:              foreach (var t in hash)
 59:              {
 60:                  sb.Append(t.ToString("X2"));
 61:              }
 62:              return sb.ToString();
 63:          }
 64:
 65:          /// <summary>
 66:          /// sifreyi kriterlere göre kontrol edip
 67:          /// uymayan kosullarini string listesine geri dönüs yapan methoddur.
 68:          /// Uygun olmasi durumunda null döndürecektir.
 69:          /// </summary>
 70:          /// <param name="text">Kullanici Sifresi</param>
 71:          /// <returns>Sifrenin Uygun Olup Olmadigi Bilgisi (Açiklamasi ile
beraber)</returns>
 72:          public static string SifreUygunMu(string text)
 73:          {
 74:              //bu kisimda sifrenin olmasi gereken kosullari degerlendirilecek.
 75:              var mesajlar = new List<string>();
 76:
 77:              if (text.Length < 8)
 78:                  mesajlar.Add("Parola en az 8 karakter olmalidir.");
 79:
 80:              if (!Regex.IsMatch(text, @"^(?=.*[a-z])", RegexOptions.ECMAScript))
 81:                  mesajlar.Add("Parola en az bir küçük harf içermelidir.");
 82:
 83:              if (!Regex.IsMatch(text, @"^(?=.*[A-Z])", RegexOptions.ECMAScript))
 84:                  mesajlar.Add("Parola en az bir büyük harf içermelidir.");
 85:
 86:              if (!Regex.IsMatch(text, @"^(?=.*[!,@,#,$,%,^,&,*,?,_,~,-,£,(,)])",
RegexOptions.ECMAScript))
 87:                  mesajlar.Add("Parola en az bir noktalama isareti içermelidir.");
 88:
 89:              if (!Regex.IsMatch(text, @"^(?=.*[0-9])", RegexOptions.ECMAScript))
 90:                  mesajlar.Add("Parola en az bir rakam içermelidir.");
 91:
 92:              if (mesajlar.Count < 1)
 93:                  return "Uygun";
 94:              var sonuc = mesajlar.Aggregate("", (current, item) => current + item + ",
");
 95:              sonuc = sonuc.Substring(0, sonuc.Length - 2);
 96:              return sonuc;
 97:          }
 98:
 99:          /// <summary>
100:          /// 8 karakter uzunlugunda Rastgele sifre olusturur.
101:          /// </summary>
102:          /// <returns>Sifrelenmis Metin Bilgisi</returns>
103:          public static string RastgeleSifreUret()
104:          {
105:              var sifreUzunluk = 8;
106:              const string gecerliKarakterler =
"abcdefghijklmnozABCDEFGHIJKLMNOZ1234567890";
107:              var sifre = new StringBuilder(100);
108:              var random = new Random();
109:              while (0 < sifreUzunluk--)
110:              {
111:
sifre.Append(gecerliKarakterler[random.Next(gecerliKarakterler.Length)]);
112:              }
```

```
113:                    return sifre.ToString();
114:                }
115:        }
116: }
```

Namespaces

Name	Description
CSharpHelperMethods (⊠ see page 1)	This is namespace CSharpHelperMethods.

1.2.20 SifrelemeIslemleriTest.cs

This is file SifrelemeIslemleriTest.cs.

Body Source

```
 1: ?using System;
 2: using CSharpHelperMethods.Library;
 3: using NUnit.Framework;
 4:
 5: namespace CSharpHelperMethods.Test
 6: {
 7:     /// <summary>
 8:     /// Library kisminda Sifreleme Islemleri bölümünde
 9:     /// Yer alan metotlarin örnek test durumlari ve beklenen sonuçlari ile
10:     /// Test edildikleri siniftir
11:     /// </summary>
12:     [TestFixture]
13:     public class SifrelemeIslemleriTest
14:     {
15:         [Test]
16:         [TestCase("1", ExpectedResult = "C4CA4238A0B923820DCC509A6F75849B")]
17:         [TestCase("serdar", ExpectedResult = "11A37959E906860A34392F586E544FD3")]
18:         public string HesaplaMD5Test(string metin)
19:         {
20:             return SifrelemeIslemleri.HesaplaMD5(metin);
21:         }
22:
23:         [Test]
24:         [TestCase("deneme", ExpectedResult =
"25B80B3556CA3A15353DD2FD312062FAD27ADCF5A1DE51B75BDADEA1FA8214AB")]
25:         [TestCase("12", ExpectedResult =
"6B51D431DF5D7F141CBECECCF79EDF3DD861C3B4069F0B11661A3EEFACBBA918")]
26:         public string HesaplaSHA256Test(string metin)
27:         {
28:             return SifrelemeIslemleri.HesaplaSHA256(metin);
29:         }
30:
31:         [Test]
32:         [TestCase("34", ExpectedResult =
"E5397F14C44F8DF754617194051DAB1AD38F59F08580406C2EFD59AA4C0F71616713C2ABE76BC503E08F2F5EDA4
863634F6FE99AD39D46C947C09623B91E53CA")]
33:         [TestCase("test", ExpectedResult =
"EE26B0DD4AF7E749AA1A8EE3C10AE9923F618980772E473F8819A5D4940E0DB27AC185F8A0E1D5F84F88BC887FD
67B143732C304CC5FA9AD8E6F57F50028A8FF")]
34:         public string HesaplaSHA512Test(string metin)
35:         {
36:             return SifrelemeIslemleri.HesaplaSHA512(metin);
37:         }
38:
39:         [Test]
40:         [TestCase("1aS_", ExpectedResult = "Parola en az 8 karakter olmalidir.")]
41:         [TestCase("56Serdar", ExpectedResult = "Parola en az bir noktalama isareti
içermelidir.")]
42:         [TestCase("54serdar_", ExpectedResult = "Parola en az bir büyük harf
```

```
içermelidir.")]
43:         [TestCase("12SERADR_", ExpectedResult = "Parola en az bir küçük harf
içermelidir.")]
44:         [TestCase("SerdarGul_", ExpectedResult = "Parola en az bir rakam içermelidir.")]
45:         [TestCase("SerdarGul1_", ExpectedResult = "Uygun")]
46:         public string SifreUygunMuTest(string metin)
47:         {
48:             return SifrelemeIslemleri.SifreUygunMu(metin);
49:         }
50:     }
51: }
```

Namespaces

Name	Description
CSharpHelperMethods (⊡ see page 1)	This is namespace CSharpHelperMethods.

1.2.21 SiralamaIslemleri.cs

This is file SiralamaIslemleri.cs.

Body Source

```
1: ?using System;
2: using System.Globalization;
3: using System.Text;
4: using static System.String;
5:
6: namespace CSharpHelperMethods.Library
7: {
8:     /// <summary>
9:     /// Alfa numerik siralama için hazirlanmis bir metod
10:    /// </summary>
11:    public static class SiralamaIslemleri
12:    {
13:        private enum ChunkType { Alphanumeric, Numeric }
14:
15:        private static bool InChunk(char ch, char otherCh)
16:        {
17:            var type = ChunkType.Alphanumeric;
18:
19:            if (char.IsDigit(otherCh))
20:            {
21:                type = ChunkType.Numeric;
22:            }
23:
24:            return (type != ChunkType.Alphanumeric || !char.IsDigit(ch)) && (type !=
ChunkType.Numeric || char.IsDigit(ch));
25:        }
26:
27:        public static int Compare(string x, string y)
28:        {
29:            if (IsNullOrEmpty(x) || IsNullOrEmpty(y)) return 0;
30:            var thisMarker = 0;
31:            var thatMarker = 0;
32:
33:            while (thisMarker < x.Length || thatMarker < y.Length)
34:            {
35:                if (thisMarker >= x.Length)
36:                {
37:                    return -1;
38:                }
39:                if (thatMarker >= y.Length)
40:                {
41:                    return 1;
```

```
 42:                    }
 43:
 44:                    var thisCh = x[thisMarker];
 45:                    var thatCh = y[thatMarker];
 46:
 47:                    var thisChunk = new StringBuilder();
 48:                    var thatChunk = new StringBuilder();
 49:
 50:                    while (thisMarker < x.Length && (thisChunk.Length == 0 ||
InChunk(thisCh, thisChunk[0])))
 51:                    {
 52:                        thisChunk.Append(thisCh);
 53:                        thisMarker++;
 54:
 55:                        if (thisMarker < x.Length)
 56:                        {
 57:                            thisCh = x[thisMarker];
 58:                        }
 59:                    }
 60:
 61:                    while (thatMarker < y.Length && (thatChunk.Length == 0 ||
InChunk(thatCh, thatChunk[0])))
 62:                    {
 63:                        thatChunk.Append(thatCh);
 64:                        thatMarker++;
 65:
 66:                        if (thatMarker < y.Length)
 67:                        {
 68:                            thatCh = y[thatMarker];
 69:                        }
 70:                    }
 71:
 72:                    var result = 0;
 73:                    // If both chunks contain numeric characters, sort them numerically
 74:                    if (char.IsDigit(thisChunk[0]) && char.IsDigit(thatChunk[0]))
 75:                    {
 76:                        var thisNumericChunk = Convert.ToInt64(thisChunk.ToString());
 77:                        var thatNumericChunk = Convert.ToInt64(thatChunk.ToString());
 78:
 79:                        if (thisNumericChunk < thatNumericChunk)
 80:                        {
 81:                            result = -1;
 82:                        }
 83:
 84:                        if (thisNumericChunk > thatNumericChunk)
 85:                        {
 86:                            result = 1;
 87:                        }
 88:                    }
 89:                    else
 90:                    {
 91:                        var ci = new CultureInfo("tr-TR");
 92:                        result = String.Compare(thisChunk.ToString(),
thatChunk.ToString(), true,
 93:                            ci);
 94:                        //result = thisChunk.ToString().CompareTo(thatChunk.ToString());
 95:                    }
 96:
 97:                    if (result == 0) continue;
 98:                    return result;
 99:                }
100:            return 0;
101:        }
102:    }
103: }
```

Namespaces

Name	Description
CSharpHelperMethods (☐ see page 1)	This is namespace CSharpHelperMethods.

1.2.22 **TarihIslemleri.cs**

This is file TarihIslemleri.cs.

Body Source

```
 1: ?using System;
 2: using System.Text;
 3:
 4: namespace CSharpHelperMethods.Library
 5: {
 6:     /// <summary>
 7:     /// Tarihsel veri kontrol metotlarini barindiran sinif
 8:     /// </summary>
 9:     public static class TarihIslemleri
10:     {
11:         /// <summary>
12:         /// Filtre ekranlarinda baslangiç ve bitis tarihi üzerinden sorgulama
yapildiginda
13:         /// Eger ayni gün için sorgulama yapiliyorsa
14:         /// Bitis tarihini o günün 23.59.59.999'una ayarlamak için
15:         /// Hazirlanan metottur
16:         /// </summary>
17:         /// <param name="dateTime">Tarih Bilgisi</param>
18:         /// <param name="hours">Saat Bilgisi</param>
19:         /// <param name="minutes">Dakika Bilgisi</param>
20:         /// <param name="seconds">Saniye Bilgisi</param>
21:         /// <param name="milliseconds">Mili Saniye Bilgisi</param>
22:         /// <returns>Yeni Tarih Bilgisi</returns>
23:         public static DateTime ZamaniDegistir(this DateTime dateTime, int hours, int
minutes, int seconds, int milliseconds)
24:         {
25:             return new DateTime(
26:                 dateTime.Year,
27:                 dateTime.Month,
28:                 dateTime.Day,
29:                 hours,
30:                 minutes,
31:                 seconds,
32:                 milliseconds,
33:                 dateTime.Kind);
34:         }
35:
36:         /// <summary>
37:         /// Yas hesaplamak için kullanilan metottur
38:         /// </summary>
39:         /// <param name="dogumTarihi">Dogum Tarihi Bilgisi</param>
40:         /// <returns>Yas Bilgisi</returns>
41:         public static int YasHesapla(DateTime dogumTarihi)
42:         {
43:             var now = DateTime.Now;
44:             var years = new DateTime(DateTime.Now.Subtract(dogumTarihi).Ticks).Year -
1;
45:             var pastYearDate = dogumTarihi.AddYears(years);
46:             for (var i = 1; i <= 12; i++)
47:             {
48:                 if (pastYearDate.AddMonths(i) == now)
49:                     break;
50:                 if (pastYearDate.AddMonths(i) < now) continue;
```

```
 51:                        break;
 52:                    }
 53:                return years;
 54:            }
 55:
 56:            /// <summary>
 57:            /// Yas hesaplamak için kullanilan metottur
 58:            /// </summary>
 59:            /// <param name="dogumTarihi">Dogum Tarihi Bilgisi</param>
 60:            /// <returns>Yas Bilgisi (Metinsel olarak)</returns>
 61:            public static string YasHesaplaMetinsel(DateTime dogumTarihi)
 62:            {
 63:                var now = DateTime.Now;
 64:                var years = new DateTime(DateTime.Now.Subtract(dogumTarihi).Ticks).Year -
1;
 65:                var pastYearDate = dogumTarihi.AddYears(years);
 66:                var months = 0;
 67:                for (var i = 1; i <= 12; i++)
 68:                {
 69:                    if (pastYearDate.AddMonths(i) == now)
 70:                    {
 71:                        months = i;
 72:                        break;
 73:                    }
 74:
 75:                    if (pastYearDate.AddMonths(i) < now) continue;
 76:                    months = i - 1;
 77:                    break;
 78:                }
 79:                var days = now.Subtract(pastYearDate.AddMonths(months)).Days;
 80:                // var hours = now.Subtract(pastYearDate).Hours;
 81:                // var minutes = now.Subtract(pastYearDate).Minutes;
 82:                // var seconds = now.Subtract(pastYearDate).Seconds;
 83:                return $"Yas: {years} Yil {months} Ay {days} Gün";
 84:                //return String.Format("Yas: {0} Yil {1} Ay {2} Gün {3} Saat {4} Saniye",
 85:                //years, months, Days, Hours, Seconds);
 86:            }
 87:
 88:            /// <summary>
 89:            /// Iki tarih arasinda kaç gün oldugunu hesaplamak için kullanilan metottur
 90:            /// </summary>
 91:            /// <param name="baslangicTarihi">Baslangiç Tarihi Bilgisi</param>
 92:            /// <param name="bitisTarihi">Bitis Tarihi Bilgisi</param>
 93:            /// <returns>Tarih Araligi Bilgisi</returns>
 94:            public static string TarihAraligiHesapla(DateTime baslangicTarihi, DateTime
bitisTarihi)
 95:            {
 96:                var years = new DateTime(bitisTarihi.Subtract(baslangicTarihi).Ticks).Year
- 1;
 97:                var pastYearDate = baslangicTarihi.AddYears(years);
 98:                var months = 0;
 99:                for (var i = 1; i <= 12; i++)
100:                {
101:                    if (pastYearDate.AddMonths(i) == bitisTarihi)
102:                    {
103:                        months = i;
104:                        break;
105:                    }
106:
107:                    if (pastYearDate.AddMonths(i) < bitisTarihi) continue;
108:                    months = i - 1;
109:                    break;
110:                }
111:                var days = bitisTarihi.Subtract(pastYearDate.AddMonths(months)).Days;
112:                var result = new StringBuilder();
113:                result.Append(years.ToString().PadLeft(2, '0'));
114:                result.Append(" Yil, ");
115:                result.Append(months.ToString().PadLeft(2, '0'));
116:                result.Append(" Ay, ");
```

1

```
117:              result.Append(days.ToString().PadLeft(2, '0'));
118:              result.Append(" Gün");
119:              return result.ToString();
120:          }
121:
122:          /// <summary>
123:          /// Kidem Tazminati Hesaplamasi Yapilirken ayda kaç gün olursa olsun tamaminda
çalismissa
124:          /// 30 gün üzerinden hesaplanmasi gerektigi için
125:          /// Isten ayrilis ve is baslangiç tarihleri arasinda kaç gün geçtigini
belirtmek için
126:          /// Hesaplama yapilmasini saglayan metottur.
127:          /// </summary>
128:          /// <param name="baslangicTarihi">Baslangiç Tarihi Bilgisi</param>
129:          /// <param name="bitisTarihi">Bitis Tarihi Bilgisi</param>
130:          /// <returns>Gün Sayisi Bilgisi</returns>
131:          public static int KidemTazminatiTarihAraligiHesapla(DateTime? baslangicTarihi,
DateTime? bitisTarihi)
132:          {
133:              var artacakMi = true;
134:              var result = 0;
135:              var baslangicMi = false;
136:              var bitisMi = false;
137:              var aylikGunSayisi = 0;
138:              for (var date = (DateTime)baslangicTarihi; date <= (DateTime)bitisTarihi;
date = date.AddDays(1))
139:              {
140:                  if (!artacakMi)
141:                      result++;
142:                  else
143:                      artacakMi = false;
144:
145:                  if (date.Day == 1)
146:                      baslangicMi = true;
147:                  else if (date.Day == DateTime.DaysInMonth(date.Year, date.Month))
148:                      bitisMi = true;
149:
150:                  if (baslangicMi)
151:                      aylikGunSayisi++;
152:                  if (baslangicMi && bitisMi && aylikGunSayisi != 30)
153:                  {
154:                      result = result - aylikGunSayisi + 30;
155:                      baslangicMi = false;
156:                      bitisMi = false;
157:                      aylikGunSayisi = 0;
158:                  }
159:                  else if (bitisMi)
160:                  {
161:                      baslangicMi = false;
162:                      bitisMi = false;
163:                      aylikGunSayisi = 0;
164:                  }
165:
166:              }
167:              if (baslangicTarihi.Value.Day > bitisTarihi.Value.Day)
168:                  result--;
169:              return result;
170:          }
171:
172:          /// <summary>
173:          /// Girilen iki tarih arasinda kaç ay oldugunu bulan metottur
174:          /// </summary>
175:          /// <param name="lValue">Ilk tarih bilgisi</param>
176:          /// <param name="rValue">Son tarih bilgisi</param>
177:          /// <returns>Ay Farki Bilgisi</returns>
178:          public static int AyFarki(DateTime lValue, DateTime rValue) =>
Math.Abs(lValue.Month - rValue.Month + 12 * (lValue.Year - rValue.Year));
179:      }
180: }
```

Namespaces

Name	Description
CSharpHelperMethods (▣ see page 1)	This is namespace CSharpHelperMethods.

1.2.23 **TarihIslemleriTest.cs**

This is file TarihIslemleriTest.cs.

Body Source

```
 1: ?using System;
 2: using CSharpHelperMethods.Library;
 3: using NUnit.Framework;
 4:
 5: namespace CSharpHelperMethods.Test
 6: {
 7:     /// <summary>
 8:     /// Library kisminda Tarih Islemleri bölümünde
 9:     /// Yer alan metotlarin örnek test durumlari ve beklenen sonuçlari ile
10:     /// Test edildikleri siniftir
11:     /// </summary>
12:     [TestFixture]
13:     public class TarihIslemleriTest
14:     {
15:         [Test]
16:         [TestCase(1987, ExpectedResult = 36)]
17:         [TestCase(2023, ExpectedResult = 0)]
18:         public int YasHesaplaTest(int yil)
19:         {
20:             var tarih = new DateTime(yil, 1, 1);
21:             return TarihIslemleri.YasHesapla(tarih);
22:         }
23:
24:         [Test]
25:         [TestCase(1987, 3, 7, ExpectedResult = "Yas: 36 Yil 0 Ay 0 Gün")]
26:         [TestCase(2022, 1, 7, ExpectedResult = "Yas: 1 Yil 2 Ay 0 Gün")]
27:         public string YasHesaplaMetinselTest(int yil, int ay, int gun)
28:         {
29:             var tarih = new DateTime(yil, ay, gun);
30:             return TarihIslemleri.YasHesaplaMetinsel(tarih);
31:         }
32:
33:         [Test]
34:         [TestCase(1987, 3, 7, ExpectedResult = "36 Yil, 00 Ay, 00 Gün")]
35:         [TestCase(2022, 1, 7, ExpectedResult = "01 Yil, 02 Ay, 00 Gün")]
36:         public string TarihAraligiHesaplaTest(int yil, int ay, int gun)
37:         {
38:             var tarih = new DateTime(yil, ay, gun);
39:             return TarihIslemleri.TarihAraligiHesapla(tarih, DateTime.Now);
40:         }
41:
42:         [Test]
43:         [TestCase(1987, 3, 7, ExpectedResult = 12961)]
44:         [TestCase(2022, 1, 7, ExpectedResult = 421)]
45:         public int KidemTazminatiTarihAraligiHesapla(int yil, int ay, int gun)
46:         {
47:             var tarih = new DateTime(yil, ay, gun);
48:             return TarihIslemleri.KidemTazminatiTarihAraligiHesapla(tarih,
DateTime.Now);
49:         }
50:
51:         [Test]
52:         [TestCase(1987, 3, 7, ExpectedResult = 432)]
```

```
53:            [TestCase(2022, 1, 7, ExpectedResult = 14)]
54:            public int AyFarkiTest(int yil, int ay, int gun)
55:            {
56:                var tarih = new DateTime(yil, ay, gun);
57:                return TarihIslemleri.AyFarki(tarih, DateTime.Now);
58:            }
59:        }
60: }
```

Namespaces

Name	Description
CSharpHelperMethods (⊡ see page 1)	This is namespace CSharpHelperMethods.

Index

D

F

T