

# C Sharp Yardimci Metotlar

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

# Table of Contents

<b>Symbol Reference</b>	<b>1</b>
<b>CSharpHelperMethods Namespace</b>	<b>1</b>
CSharpHelperMethods.Properties Namespace	1
Classes	1
Resources Class	1
Settings Class	3
CSharpHelperMethods.YardimciSiniflar Namespace	4
Classes	4
DataTableIslemleri Class	5
DosyaIslemleri Class	7
KisiIslemleri Class	11
MetinIslemleri Class	13
SayiIslemleri Class	19
SifrelemeliIslemleri Class	22
SiralamaIslemleri Class	25
TarihIslemleri Class	27
Classes	32
FrmAnaSayfa Class	32
FrmAnaSayfa.FrmAnaSayfa Constructor	34
FrmAnaSayfa Fields	34
FrmAnaSayfa Methods	42
Program Class	62
Program Methods	62
<b>Files</b>	<b>62</b>
AssemblyInfo.cs	63
CSharpHelperMethods.csproj	64
CSharpHelperMethods.sln	64
DataTableIslemleri.cs	64
DosyaIslemleri.cs	66
FrmAnaSayfa.cs	68
FrmAnaSayfa.Designer.cs	78
KisiIslemleri.cs	89
MetinIslemleri.cs	91
Program.cs	94
Resources.Designer.cs	94
SayiIslemleri.cs	96

Settings.Designer.cs	98
Sifrelemelslemleri.cs	99
Siralamalslemleri.cs	101
TarihIslemleri.cs	102

## Index a

# 1 Symbol Reference



## 1.1 CSharpHelperMethods Namespace

This is namespace CSharpHelperMethods.

### Namespaces

Name	Description
Properties ( <a href="#">see page 1</a> )	This is namespace CSharpHelperMethods.Properties.
YardimciSiniflar ( <a href="#">see page 4</a> )	This is namespace CSharpHelperMethods.YardimciSiniflar.



### Classes

	Name	Description
	FrmAnaSayfa ( <a href="#">see page 32</a> )	This is class CSharpHelperMethods.FrmAnaSayfa.
	Program ( <a href="#">see page 62</a> )	This is class CSharpHelperMethods.Program.

### 1.1.1 CSharpHelperMethods.Properties Namespace

This is namespace CSharpHelperMethods.Properties.



### Classes

	Name	Description
	Resources ( <a href="#">see page 1</a> )	A strongly-typed resource class, for looking up localized strings, etc.
	Settings ( <a href="#">see page 3</a> )	This is class CSharpHelperMethods.Properties.Settings.

#### 1.1.1.1 Classes

The following table lists classes in this documentation.

### Classes

	Name	Description
	Resources ( <a href="#">see page 1</a> )	A strongly-typed resource class, for looking up localized strings, etc.
	Settings ( <a href="#">see page 3</a> )	This is class CSharpHelperMethods.Properties.Settings.

#### 1.1.1.1.1 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.StronglyTypedResourceBuilder",
"4.0.0.0")]
[global::System.Diagnostics.DebuggerNonUserCodeAttribute()]
[global::System.Runtime.CompilerServices.CompilerGeneratedAttribute()]
internal class Resources;
```


**File**

Resources.Designer.cs (see page 94)



**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 2)	This is Resources, a member of class Resources.

**Resources Fields**

	Name	Description
	resourceCulture (see page 2)	This is resourceCulture, a member of class Resources.
	resourceMan (see page 3)	This is resourceMan, a member of class Resources.

**Resources Properties**

	Name	Description
	Culture (see page 3)	Overrides the current thread's CurrentUICulture property for all resource lookups using this strongly typed resource class.
	ResourceManager (see page 3)	Returns the cached ResourceManager instance used by this class.

**1.1.1.1.1.1 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: }
```

**1.1.1.1.1.2 Resources Fields****1.1.1.1.1.2.1 Resources.resourceCulture Field****C#**

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

**Description**

This is resourceCulture, a member of class Resources.

#### 1.1.1.1.2.2 Resources.resourceMan Field

C#

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

##### Description

This is resourceMan, a member of class Resources.

#### 1.1.1.1.3 Resources Properties

##### 1.1.1.1.3.1 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.EditorBrowsableState.Advanced)]  
internal static global::System.Globalization.CultureInfo Culture;
```

##### 1.1.1.1.3.2 Resources.ResourceManager Property

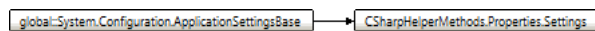
Returns the cached ResourceManager instance used by this class.

C#

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

#### 1.1.1.1.2 Settings Class

##### Class Hierarchy



C#

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

##### File

Settings.Designer.cs (see page 98)

##### Description

This is class CSharpHelperMethods.Properties.Settings.

##### Settings Fields

	Name	Description
	defaultInstance (see page 4)	This is defaultInstance, a member of class Settings.

##### Settings Properties

	Name	Description
	Default (see page 4)	This is Default, a member of class Settings.

#### 1.1.1.1.2.1 Settings Fields

#### 1.1.1.1.2.1.1 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.

#### 1.1.1.1.2.2 Settings Properties

##### 1.1.1.1.2.2.1 Settings.Default Property

C#

```
public static Settings Default;
```









#### Description

This is Default, a member of class Settings.

## 1.1.2 CSharpHelperMethods.YardimciSiniflar Namespace

This is namespace CSharpHelperMethods.YardimciSiniflar.







#### Classes



	Name	Description
	DataTablesIslemleri ( <a href="#">see page 5</a> )	Raporlama için lazım olan DataTable vb. verilerin düzenlenmesini sağlayan sınıflardır.
	DosyaIslemleri ( <a href="#">see page 7</a> )	Dosya işlemleri ile ilgili yardımcı sınıf
	KisiIslemleri ( <a href="#">see page 11</a> )	Kisi ile ilişkili yardımcı metotların tamamını içeren sınıf
	MetinIslemleri ( <a href="#">see page 13</a> )	Metinsel veri kontrol metotlarını barındıran sınıf
	SayıIslemleri ( <a href="#">see page 19</a> )	Sayısal veri kontrol metotlarını barındıran sınıf
	Sifrelemelslemleri ( <a href="#">see page 22</a> )	Sifre ile ilgili işlemler sınıfı
	Siralamalslemleri ( <a href="#">see page 25</a> )	Alfa numerik sıralama için hazırlanmış bir metod
	TarihIslemleri ( <a href="#">see page 27</a> )	Tarihsel veri kontrol metotlarını barındıran sınıf

### 1.1.2.1 Classes

The following table lists classes in this documentation.

#### Classes

	Name	Description
	DataTablesIslemleri ( <a href="#">see page 5</a> )	Raporlama için lazım olan DataTable vb. verilerin düzenlenmesini sağlayan sınıflardır.
	DosyaIslemleri ( <a href="#">see page 7</a> )	Dosya işlemleri ile ilgili yardımcı sınıf
	KisiIslemleri ( <a href="#">see page 11</a> )	Kisi ile ilişkili yardımcı metotların tamamını içeren sınıf
	MetinIslemleri ( <a href="#">see page 13</a> )	Metinsel veri kontrol metotlarını barındıran sınıf
	SayıIslemleri ( <a href="#">see page 19</a> )	Sayısal veri kontrol metotlarını barındıran sınıf
	Sifrelemelslemleri ( <a href="#">see page 22</a> )	Sifre ile ilgili işlemler sınıfı

	SiralamaIslemleri (see page 25)	Alfa numerik siralama için hazirlanmis bir metod
	TarihIslemleri (see page 27)	Tarihsel veri kontrol metotlarini barindiran sinif

### 1.1.2.1.1 DataTableIslemleri Class

Raporlama için lazım olan DataTable vb. verilerin düzenlenmesini sağlayan siniflardir.

#### Class Hierarchy

CSharpHelperMethods.YardimciSiniflar.DataTableIslemleri






#### C#

```
public static class DataTableIslemleri;
```

#### File

DataTableIslemleri.cs (see page 64)

#### DataTableIslemleri Methods

	Name	Description
	BaseType (see page 5)	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 5)	This is Nullable.GetUnderlyingType, a member of class DataTableIslemleri.
	oType.GetGenericTypeDefinition (see page 6)	This is oType.GetGenericTypeDefinition, a member of class DataTableIslemleri.
	ToDataTable (see page 6)	Parametre olarak girilen herhangi bir sinif türünden listeyi Idael bir sekilde Nullable alan içermeyen bir DataTable'a çevirmek için kullanılan metottur.
	typeof (see page 7)	This is typeof, a member of class DataTableIslemleri.

### 1.1.2.1.1.1 DataTableIslemleri Methods

#### 1.1.2.1.1.1.1 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

#### 1.1.2.1.1.1.2 DataTableIslemleri.Nullable.GetUnderlyingType Method

#### C#

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

#### Description

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



### 1.1.2.1.1.3 DataTableIslemleri.oType.GetGenericTypeDefinition Method

C#

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

#### Description

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

### 1.1.2.1.1.4 DataTableIslemleri.ToDataTable Method

Parametre olarak girilen herhangi bir sinif türünden listeyi Idael bir sekilde Nullable alan içermeyen bir DataTable'a çevirmek için kullanılan 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:     sonuc.Rows.Add(degerler);
63: }
64:
65: return sonuc;
66: }
67: catch (Exception)
68: {
69:     return null;
70: }
71: }
72: }

```

#### 1.1.2.1.1.5 DataTableIslemleri.typeof Method

**C#**

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

#### Description

This is typeof, a member of class DataTableIslemleri.

#### 1.1.2.1.2 Dosyaislemleri Class

Dosya islemleri ile ilgili yardimci sinif

#### Class Hierarchy

```
CSharpHelperMethods.YardimciSiniflar.Dosyaislemleri
```






**C#**

```
public static class DosyaIslemleri;
```

**File**

DosyaIslemleri.cs (see page 66)

**DosyaIslemleri Methods**

	Name	Description
	DosyaKilitliMi (see page 8)	Dosya sistemindeki bir dosyanin baska bir program tarafından o anda kullanimda olup olmadiginin kontrol edildiği bir metottur.
	DosyaUzantisiUygunMu (see page 9)	Sisteme yüklenecek dosyanin belirlenen kriterlere uyup uymadigini kontrol eden metottur. Dosyanin uzantisi istedigimiz formatta degilse yanlis olarak geri döndürecektr.s
	GetirDosyaUzantisiMimeTypelle (see page 9)	Dosyanin türünden dosyanin uzantisinin ne oldugunu getiren metottur
	GetirIcerikTipiDosyaUzantisille (see page 10)	Dosya uzantisinden dosyanin türünün ne oldugunu getiren metottur
	GüvenliDizinOlustur (see page 10)	Bu metot ile gönderdiginiz Dizin yolunda verdiginiz isimde bir klasör yoksa Otomatik olarak olusturmayi sagliyor

**1.1.2.1.2.1 DosyaIslemleri Methods****1.1.2.1.2.1.1 DosyaIslemleri.DosyaKilitliMi Method**

Dosya sistemindeki bir dosyanin baska bir program tarafından o anda kullanimda olup olmadiginin kontrol edildiği 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 tarafından kullaniliyorsa
11:         // burada true dönüyor ki
12:         // kullanıcı bu metottan gelen bilgiye göre davranisin
13:         return true;
14:     }
15:     finally
16:     {
17:         stream?.Close();
18:     }
19:
20:     //file is not locked
```

```

21:         return false;
22:     }

```

#### 1.1.2.1.2.1.2 Dosyaislemleri.DosyaUzantisiUygunMu Method

Sisteme yüklenecek dosyanın belirlenen kriterlere uyup uymadığını kontrol eden metottur. Dosyanın uzantisi istediğimiz formatta değilse yanlış olarak geri döndürecektir.

**C#**

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

**Parameters**

Parameters	Description
string extension	Dosya Uzantisi

**Returns**

Eklenecek Dosyanın Uygun Olup Olmadığı 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: }

```

#### 1.1.2.1.2.1.3 Dosyaislemleri.GetirDosyaUzantisiMimeTypeIle Method

Dosyanın türünden dosyanın uzantisinin ne olduğunu 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-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: }

```

#### 1.1.2.1.2.1.4 Dosyaislemleri.GetirIcerikTipiDosyaUzantisille Method

Dosya uzantisinden dosyanın türünün ne olduğunu getiren metottur

**C#**

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

**Parameters**

Parameters	Description
string extension	Dosya Uzantisi

**Returns**

Dosya Türü Bilgisi

**Body Source**

```

1: public static string GetirIcerikTipiDosyaUzantisille(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: }

```

#### 1.1.2.1.2.1.5 Dosyaislemleri.GuvenliDizinOlustur Method

Bu metot ile gönderdiğiniz Dizin yolunda verdiğiniz isimde bir klasör yoksa Otomatik olarak oluşturmaya sağlıyor

**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: }
```

**1.1.2.1.3 Kisilslemleri Class**

Kisi ile iliskili yardimci metotlarin tamamini içeren sinif

**Class Hierarchy**

CSharpHelperMethods.YardimciSiniflar.Kisilslemleri




**C#**

```
public static class KisiIslemleri;
```

**File**

Kisilslemleri.cs (see page 89)

**Kisilslemleri Methods**

	Name	Description
	DogrulaEPosta (see page 11)	Ekranlardan girilen bir e-posta adresinin geçerli bir e-posta adresi olup olmadigini Kontrol eden metottur
	DogrulaIBAN (see page 12)	Girilen IBAN'in standar IBAN algoritmasına uygun olup olmadigini dogrular
	DogrulaTcKimlikNo (see page 12)	Girilen TC Kimlik Numarasinin, standart TC Kimlik No algoritmasına uygun olup olmadigini kontrol eden metottur.

**1.1.2.1.3.1 Kisilslemleri Methods****1.1.2.1.3.1.1 Kisilslemleri.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: }

```

**1.1.2.1.3.1.2 Kisilslemleri.DogrulaIBAN Method**

Girilen IBAN'in standar IBAN algoritmasına 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: }

```

**1.1.2.1.3.1.3 Kisilslemleri.DogrulaTcKimlikNo Method**

Girilen TC Kimlik Numarasinin, standart TC Kimlik No algoritmasına 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: }
```

**1.1.2.1.4 MetinIslemleri Class**

Metinsel veri kontrol metotlarini barindiran sinif

**Class Hierarchy**

CSharpHelperMethods.YardimciSiniflar.MetinIslemleri








**C#**

```
public static class MetinIslemleri;
```








**File**

MetinIslemleri.cs (see page 91)

**MetinIslemleri Methods**

	Name	Description
	BoslukKaldir (see page 14)	Girilen metnin bosluklarinin kaldırilmasini saglayan metottur.
	HTMLDuzelt (see page 14)	Girilen metnin basinda virgöl olup olmadigini kontrol eden metottur.
	IdealMetinGetir (see page 15)	Girilen metnin basinda virgöl olup olmadigini kontrol eden metottur.
	MetinIlkKarakterBuyukDigerleriKucukYap (see page 16)	This is MetinIlkKarakterBuyukDigerleriKucukYap, a member of class MetinIslemleri.
	MetinIlkKarakterleriBuyukYap (see page 16)	Girilen bir metindeki tüm kelimelerin ilk harflerinin Büyük yapılması için hazirlanan metottur
	MetinSifrele (see page 16)	This is MetinSifrele, a member of class MetinIslemleri.
	MetinZararliKarakterDuzelt (see page 17)	This is MetinZararliKarakterDuzelt, a member of class MetinIslemleri.



	TelefonNoDuzenle (see page 17)	Girilen telefon numarasinin SMS gönderilecek hale getirilmesini sağlayan metottur.
	text.First (see page 17)	This is text.First, a member of class MetinIslemleri.
	text.Substring (see page 17)	This is text.Substring, a member of class MetinIslemleri.
	ToString (see page 18)	This is ToString, a member of class MetinIslemleri.
	ToUpper (see page 18)	This is ToUpper, a member of class MetinIslemleri.
	TurkceKarakterleriDuzelt (see page 18)	Girilen metnin basinda virgöl olup olmadigini kontrol eden metottur.
	UTF8Duzelt (see page 18)	Girilen metnin basinda virgöl olup olmadigini kontrol eden metottur.

#### 1.1.2.1.4.1 MetinIslemleri Methods

##### 1.1.2.1.4.1.1 MetinIslemleri.BoslukKaldir Method

Girilen metnin bosluklarinin kaldırilmasini 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 = metin.Replace("-", " ").Replace(" ", " ").Replace("(", " ").Replace(")", " ");
5:     return metin;
6: }
7: }
```

##### 1.1.2.1.4.1.2 MetinIslemleri.HTMLDuzelt Method

Girilen metnin basinda virgöl 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:
13: @"<(meta|link|?o:|/?style|/?div|/?st\d|/?head|/?html|body|/?body|/?span|!\\[\\]^>]*?>",
14:         // Get rid of empty paragraph tags
15:         @"(<[>]+>)+&nbsp;(</\\w+>)+",
16:         @"<a[>]*>([<]+)</a>",
17:         @"<p[>]*>([<]+)</p>",
18:         // remove bizarre v: element attached to <img> tag
19:         @"\\s+v:\\w+=\""[^"]*\""+",
20:         // remove extra lines
21:         @"(\\n\\r){2,}",
22:         "&nbsp;"
23:     };
24:     text = sc.Aggregate(text, (current, s) => Regex.Replace(current, s, string.Empty,
25:     RegexOptions.IgnoreCase));
26:     text = Regex.Replace(text, "<.*?>", String.Empty);
27:     text = text.Replace("\\r", " ").Replace("\\n", " ").Replace("  ", " ").Trim();
28:     text = text.Replace("&rsquo;", "'");
29:     text = text.Replace("&rsquo;", "'");
30:     text = text.Replace("&rdquo;", "\"");
31:     text = text.Replace("&rdquo;", "\"");
32:     text = text.Replace("&lsquo;", "'");
33:     text = text.Replace("&lsquo;", "'");
34:     text = text.Replace("&ldquo;", "\"");
35:     text = text.Replace("&ldquo;", "\"");
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>", "&");
41:     //text = text.Replace("&/p>", "&");
42:     //text = text.Replace("&gt;", "&");
43:     //return text;
44:     return text;
45: }

```

#### 1.1.2.1.4.1.3 MetinIslemleri.IdealMetinGetir Method

Girilen metnin basinda virgöl 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: }

```

#### 1.1.2.1.4.1.4 MetinIslemleri.MetinIlkKarakterBuyukDigerleriKucukYap Method

**C#**

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

##### Description

This is MetinIlkKarakterBuyukDigerleriKucukYap, a member of class MetinIslemleri.

#### 1.1.2.1.4.1.5 MetinIslemleri.MetinIlkKarakterleriBuyukYap Method

Girilen bir metindeki tüm kelimelerin ilk harflerinin Büyük yapılması için hazırlanan metottur

**C#**

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

##### Parameters

Parameters	Description
this string input	Girilen metin

##### Returns

Metnin Kelimelerinin İlk Harflerinin Büyülmüş Hali

##### Body Source

```
1: public static string MetinIlkKarakterleriBuyukYap(this string input)
2: {
3:     var result = "";
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:             result = result + " " + item.Substring(0, 1).ToUpper(cultureInfo) +
item.Substring(1).ToLower(cultureInfo) + " ";
10:        else if (item.Length == 1)
11:            result = result + " " + item.ToUpper(cultureInfo) + " ";
12:        }
13:        return result;
14: }
```

#### 1.1.2.1.4.1.6 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: }

```

#### 1.1.2.1.4.1.7 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: }

```

#### 1.1.2.1.4.1.8 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 = telefonNo.Replace("-", "").Replace(" ", "").Replace("(",
5:     ").Replace(")", "");
6:     if (telefonNo.Length == 11)
7:     {
8:         telefonNo = telefonNo.Substring(telefonNo.Length - 10);
9:     }
10:    return telefonNo;
11: }

```

#### 1.1.2.1.4.1.9 MetinIslemleri.text.First Method

**C#**

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

##### Description

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

#### 1.1.2.1.4.1.10 MetinIslemleri.text.Substring Method

**C#**

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

**Description**

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

**1.1.2.1.4.1.11 MetinIslemleri.ToString Method****C#**

```
private ToString();
```

**Description**

This is ToString, a member of class MetinIslemleri.

**1.1.2.1.4.1.12 MetinIslemleri.ToUpper Method****C#**

```
private ToUpper();
```

**Description**

This is ToUpper, a member of class MetinIslemleri.

**1.1.2.1.4.1.13 MetinIslemleri.TurkceKarakterleriDuzelt Method**

Girilen metnin basinda virgöl 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", "ı");
5:     text = text.Replace("ö", "o");
6:     text = text.Replace("ü", "u");
7:     text = text.Replace("s", "ş");
8:     text = text.Replace("g", "ğ");
9:     text = text.Replace("ç", "c");
10:    text = text.Replace("Ü", "U");
11:    text = text.Replace("İ", "I");
12:    text = text.Replace("Ö", "O");
13:    text = text.Replace("Ü", "U");
14:    text = text.Replace("S", "Ş");
15:    text = text.Replace("G", "Ğ");
16:    text = text.Replace("Ç", "C");
17:    return text;
18: }
```

**1.1.2.1.4.1.14 MetinIslemleri.UTF8Duzelt Method**

Girilen metnin basinda virgöl 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("&ccedil;", "ç");
4:     text = text.Replace("&ouml;", "ö");
5:     text = text.Replace("&uuml;", "ü");
6:     text = text.Replace("ç", "ç");
7:     text = text.Replace("ö", "ö");
8:     text = text.Replace("ü", "ü");
9:     return text;
10: }
```

**1.1.2.1.5 Sayislemeleri Class**

Sayisal veri kontrol metotlarini barindiran sinif

**Class Hierarchy**

CSharpHelperMethods.YardimciSiniflar.Sayislemeleri






**C#**

```
public static class SayiIslemleri;
```

**File**

Sayislemeleri.cs (see page 96)

**Sayislemeleri Methods**

	Name	Description
	AltiHaneliOnayKoduOlustur (see page 19)	Sadece rakamlardan olusan 6 haneli random sayi üreten metottur.
	SayiHalineCevir (see page 20)	Ekrandan alinan string degerde virgöl olduğundan ajax metodu patladigi için Burada string olarak tutar verisi aliniyor Ve decimala çevriliyor
	SayisalMi (see page 20)	Girilen bir metnin sayi olup olmadigini kontrol eden metottur.
	SayiyiYaziyaCevir (see page 21)	Elimizdeki tutar verilerinin Yazı formatında gösterilmesini sağlayacak bir metottur
	text.All (see page 22)	This is text.All, a member of class Sayislemeleri.

**1.1.2.1.5.1 Sayislemeleri Methods****1.1.2.1.5.1.1 Sayislemeleri.AltiHaneliOnayKoduOlustur Method**

Sadece rakamlardan olusan 6 haneli random sayi üreten metottur.

**C#**

```
public static string AltıHaneliOnayKoduOlustur();
```

**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: }

```

**1.1.2.1.5.1.2 SayiIslemleri.SayiHalineCevir Method**

Ekrandan alınan string degerde virgül olduğundan ajax metodu patladığı 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("_",
13:        ""), NumberStyles.Number,
14:        CultureInfo.InvariantCulture, out val);
15:    return value ? val : 0;

```

**1.1.2.1.5.1.3 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

#### 1.1.2.1.5.1.4 SayiIslemleri.SayiyiYaziyaCevir Method

Elimizdeki tutar verilerinin Yazı formatında gösterilmesini sağlayacak bir metottur

**C#**

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

**Parameters**

Parameters	Description
decimal tutar	Girilen Sayı Bilgisi

**Returns**

Sayının Yazıya Dönüştürülmüş Hali

**Body Source**

```
1: public static string SayiyiYaziyaCevir(decimal tutar)
2: {
3:     var sTutar = tutar.ToString("F2").Replace('.', ','); // Replace('.', ',') ondalik
    ayracinin . olma durumu için
4:     var lira = sTutar.Substring(0, sTutar.IndexOf(',')); //tutarin tam kısmi
5:     var kurus = sTutar.Substring(sTutar.IndexOf(',') + 1, 2);
6:     var yazi = new StringBuilder();
7:
8:     var birler = new[] { "", "BİR", "İKİ", "ÜÇ", "DÖRT", "BES", "ALTI", "YEDİ", "SEKİZ",
    "DOKUZ" };
9:     var onlar = new[] { "", "ON", "YIRMI", "OTUZ", "KIRK", "ELLİ", "ALTMIS", "YETMİS",
    "SEKSEN", "DOKSAN" };
10:    var binler = new[] { "KENTRİLYON", "KATRİLYON", "TRİLYON", "MİLYAR", "MİLYON",
    "BİN", "" };
11:
12:    const int grupSayisi = 7;
13:    //sayidaki 3'lü grup sayisi. katrilyon içi 6. (1.234,00 daki grup sayisi 2'dir.)
14:    //KATRİLYON'un basına ekleyeceğiniz her değer 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("BİRÜZ"))
25:            grupDegeri.Replace("BİRÜZ", "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("BİRBİN"))
31:            grupDegeri.Replace("BİRBİN", "BİN");
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: }

```

#### 1.1.2.1.5.1.5 Sayislemeleri.text.All Method

**C#**

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

#### Description

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

### 1.1.2.1.6 Sifremelslemeleri Class

Sifre ile ilgili islemler sinifi

#### Class Hierarchy

CSharpHelperMethods.YardimciSiniflar.Sifremelslemeleri






**C#**

```
public static class SifrelemeIslemleri;
```

#### File

Sifremelslemeleri.cs ([see page 99](#))

#### Sifremelslemeleri Methods

	Name	Description
	HesaplaMD5 ( <a href="#">see page 22</a> )	Kullanici sifresinin MD5 yöntemiyle sifrenmesini saglayan metottur.
	HesaplaSHA256 ( <a href="#">see page 23</a> )	Kullanici sifresinin SHA yöntemiyle sifrenmesini saglayan metottur.
	HesaplaSHA512 ( <a href="#">see page 23</a> )	Kullanici sifresinin SHA yöntemiyle sifrenmesini saglayan metottur.
	RastgeleSifreUret ( <a href="#">see page 24</a> )	8 karakter uzunlugunda Rastgele sifre olusturur.
	SifreUygunMu ( <a href="#">see page 24</a> )	sifreyi kriterlere göre kontrol edip uymayan kosullarini string listesine geri dönüş yapan methoddur. Uygun olmasi durumunda null döndürecektir.

#### 1.1.2.1.6.1 Sifremelslemeleri Methods

##### 1.1.2.1.6.1.1 Sifremelslemeleri.HesaplaMD5 Method

Kullanici sifresinin MD5 yöntemiyle sifrenmesini saglayan metottur.

**C#**

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

#### Parameters

Parameters	Description
string text	Sifrelenecek Metin

#### Returns

Girdi Metninin Sifrenmis 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: }
```

**1.1.2.1.6.1.2 Sifrelemelslemleri.HesaplaSHA256 Method**

Kullanici sifresinin SHA yöntemiyle sifrenmesini saglayan metottur.

**C#**

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

**Parameters**

Parameters	Description
string text	Sifrelenecek Metin

**Returns**

Girdi Metninin Sifrenmis 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: }
```

**1.1.2.1.6.1.3 Sifrelemelslemleri.HesaplaSHA512 Method**

Kullanici sifresinin SHA yöntemiyle sifrenmesini saglayan metottur.

**C#**

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

**Parameters**

Parameters	Description
string text	Sifrelenecek Metin

**Returns**

Girdi Metninin Sifrenmis 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: }

```

#### 1.1.2.1.6.1.4 Sifremelslemleri.RastgeleSifreUret Method

8 karakter uzunlugunda Rastgele sifre olusturur.

**C#**

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

**Returns**

Sifrenin Metin Bilgisi

**Body Source**

```

1: public static string RastgeleSifreUret()
2: {
3:     var sifreUzunluk = 8;
4:     const string gecerliKarakterler = "abcdefghijklmnopqrstuvwxyz1234567890";
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: }

```

#### 1.1.2.1.6.1.5 Sifremelslemleri.SifreUygunMu Method

sifreyi kriterlere göre kontrol edip uymayan kosullarini string listesine geri dönüş 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, @"^(?=.*[!@#%&*?_~,-,£,(,)])",
16:        RegexOptions.ECMAScript))
17:        mesajlar.Add("Parola en az bir noktalama isareti içermelidir.");
18:
19:     if (!Regex.IsMatch(text, @"^(?=.*[0-9])", RegexOptions.ECMAScript))
20:        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: }

```

### 1.1.2.1.7 Siralamalslemleri Class

Alfa numerik siralama için hazırlanmış bir metod

#### Class Hierarchy

CSharpHelperMethods.YardimciSiniflar.Siralamalslemleri


#### C#

```
public static class SiralamaIslemleri;
```



#### File

Siralamalslemleri.cs (see page 101)

#### Siralamalslemleri Enumerations

	Name	Description
	ChunkType (see page 25)	This is record CSharpHelperMethods.YardimciSiniflar.Siralamalslemleri.ChunkType.

#### Siralamalslemleri Methods

	Name	Description
	Compare (see page 25)	This is Compare, a member of class Siralamalslemleri.
	InChunk (see page 27)	This is InChunk, a member of class Siralamalslemleri.

### 1.1.2.1.7.1 Siralamalslemleri Enumerations

#### 1.1.2.1.7.1.1 CSharpHelperMethods.YardimciSiniflar.Siralamalslemleri.ChunkType Enumeration

#### C#

```

private enum ChunkType {
    Alphanumeric,
    Numeric
}

```

#### File

Siralamalslemleri.cs (see page 101)

#### Description

This is record CSharpHelperMethods.YardimciSiniflar.Siralamalslemleri.ChunkType.

### 1.1.2.1.7.2 Siralamalslemleri Methods

#### 1.1.2.1.7.2.1 Siralamalslemleri.Compare Method

#### C#

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

#### Description

This is Compare, a member of class Siralamalslemleri.

**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: }

```

#### 1.1.2.1.7.2.2 Sıralamalslemleri.InChunk Method

**C#**

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

#### Description

This is InChunk, a member of class Sıralamalslemleri.

#### 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: }

```

### 1.1.2.1.8 Tarihslemleri Class

Tarihsel veri kontrol metotlarını barındıran sınıf

#### Class Hierarchy

CSharpHelperMethods.YardimciSiniflar.Tarihslemleri

**C#**

```
public static class Tarihslemleri;
```

#### File

Tarihslemleri.cs (see page 102)

#### Tarihslemleri Methods

	Name	Description
💎💎💎	AyFarki (see page 28)	Girilen iki tarih arasında kaç ay olduğunu bulan metottur
💎💎💎	KidemTazminatıTarihAralığıHesapla (see page 28)	Kidem Tazminatı Hesaplaması Yapılırken ayda kaç gün olursa olsun tamamında çalışmıssa 30 gün üzerinden hesaplanması gerektiği için İsten ayrılış ve iş başlangıç tarihleri arasında kaç gün geçtiğini belirtmek için Hesaplama yapılmasını sağlayan metottur.
💎💎💎	Math.Abs (see page 29)	This is Math.Abs, a member of class Tarihslemleri.
💎💎💎	TarihAralığıHesapla (see page 29)	İki tarih arasında kaç gün olduğunu hesaplamak için kullanılan metottur
💎💎💎	YasHesapla (see page 30)	Yas hesaplamak için kullanılan metottur
💎💎💎	YasHesaplaMetinsel (see page 30)	Yas hesaplamak için kullanılan metottur

	ZamaniDegistir (see page 31)	Filtre ekranlarında başlangıç ve bitis tarihi üzerinden sorgulama yapıldığında Eger aynı gün için sorgulama yapılıyorsa Bitis tarihini o günün 23.59.59.999'una ayarlamak için Hazırlanan metottur
--	------------------------------	--

### 1.1.2.1.8.1 TarihIslemleri Methods

#### 1.1.2.1.8.1.1 TarihIslemleri.AyFarki Method

Girilen iki tarih arasında kaç ay olduğunu bulan metottur

**C#**

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

**Parameters**

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

**Returns**

Ay Farki Bilgisi

#### 1.1.2.1.8.1.2 TarihIslemleri.KidemTazminatıTarihAralığıHesapla Method

Kidem Tazminatı Hesaplaması Yapılırken ayda kaç gün olursa olsun tamamında çalışmıssa 30 gün üzerinden hesaplanması gerektiği için İsten ayrılış ve iş başlangıç tarihleri arasında kaç gün geçtiğini belirtmek için Hesaplama yapılmasını sağlayan metottur.

**C#**

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

**Parameters**

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

**Returns**

Gün Sayısı Bilgisi

**Body Source**

```
1: public static int KidemTazminatıTarihAralığıHesapla(DateTime? baslangicTarihi,
2: DateTime? bitisTarihi)
3: {
4:     var artacakMi = true;
5:     var result = 0;
6:     var baslangicMi = false;
7:     var bitisMi = false;
8:     var aylıkGünSayısı = 0;
9:     for (var date = (DateTime)baslangicTarihi; date <= (DateTime)bitisTarihi; date =
10: date.AddDays(1))
11:     {
12:         if (!artacakMi)
13:             result++;
14:         else
15:             artacakMi = false;
16:         if (date.Day == 1)
17:             baslangicMi = true;
18:         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: }

```

#### 1.1.2.1.8.1.3 Tarihslemleri.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 Tarihslemleri.

#### 1.1.2.1.8.1.4 Tarihslemleri.TarihAraligiHesapla Method

İki tarih arasında kaç gün olduğunu hesaplamak için kullanılan metottur

C#

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

#### Parameters

Parameters	Description
DateTime baslangicTarihi	Baslangıç 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: }

```

#### 1.1.2.1.8.1.5 Tarihslemleri.YasHesapla Method

Yas hesaplamak için kullanılan metottur

**C#**

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

**Parameters**

Parameters	Description
DateTime dogumTarihi	Dogum Tarihi Bilgisi

**Returns**

Yas Bilgisi

**Body Source**

```

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:     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:    return years;
20: }

```

#### 1.1.2.1.8.1.6 Tarihslemleri.YasHesaplaMetinsel Method

Yas hesaplamak için kullanılan metottur

**C#**

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

**Parameters**

Parameters	Description
DateTime dogumTarihi	Dogum Tarihi Bilgisi

**Returns**

Yas Bilgisi (Metinsel olarak)

**Body Source**

```

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: }

```

**1.1.2.1.8.1.7 Tarihslemleri.ZamaniDegistir Method**

Filtre ekranlarında başlangıç ve bitis tarihi üzerinden sorgulama yapıldığında Eger aynı gün için sorgulama yapılıyorsa Bitis tarihini o günün 23.59.59.999'una ayarlamak için Hazirlanan metottur

**C#**

```

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**

```



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: }

```

## 1.1.3 Classes

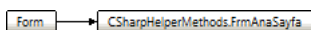
The following table lists classes in this documentation.

### Classes

	Name	Description
	FrmAnaSayfa ( <a href="#">see page 32</a> )	This is class CSharpHelperMethods.FrmAnaSayfa.
	Program ( <a href="#">see page 62</a> )	This is class CSharpHelperMethods.Program.

### 1.1.3.1 FrmAnaSayfa Class

#### Class Hierarchy



#### C#

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


#### File

FrmAnaSayfa.Designer.cs ([see page 78](#))
















#### Description

This is class CSharpHelperMethods.FrmAnaSayfa.

#### Methods





	Name	Description
	FrmAnaSayfa ( <a href="#">see page 34</a> )	This is FrmAnaSayfa, a member of class FrmAnaSayfa.

#### FrmAnaSayfa Fields

	Name	Description
	btnAyFarki ( <a href="#">see page 34</a> )	This is btnAyFarki, a member of class FrmAnaSayfa.
	btnEPostaDogrula ( <a href="#">see page 34</a> )	This is btnEPostaDogrula, a member of class FrmAnaSayfa.
	btnIbanDogrula ( <a href="#">see page 34</a> )	This is btnIbanDogrula, a member of class FrmAnaSayfa.
	btnIlkHarfiBuyukYap ( <a href="#">see page 35</a> )	This is btnIlkHarfiBuyukYap, a member of class FrmAnaSayfa.
	btnIlkHarfleriBuyukYap ( <a href="#">see page 35</a> )	This is btnIlkHarfleriBuyukYap, a member of class FrmAnaSayfa.
	btnMd5Hash ( <a href="#">see page 35</a> )	This is btnMd5Hash, a member of class FrmAnaSayfa.
	btnMetinSifrele ( <a href="#">see page 35</a> )	This is btnMetinSifrele, a member of class FrmAnaSayfa.
	btnMetneDonustur ( <a href="#">see page 35</a> )	This is btnMetneDonustur, a member of class FrmAnaSayfa.
	btnOnayKoduOlustur ( <a href="#">see page 35</a> )	This is btnOnayKoduOlustur, a member of class FrmAnaSayfa.
	btnRastgeleSifreUret ( <a href="#">see page 35</a> )	This is btnRastgeleSifreUret, a member of class FrmAnaSayfa.
	btnSayisalMi ( <a href="#">see page 36</a> )	This is btnSayisalMi, a member of class FrmAnaSayfa.
	btnSha256Hash ( <a href="#">see page 36</a> )	This is btnSha256Hash, a member of class FrmAnaSayfa.
	btnSha512Hash ( <a href="#">see page 36</a> )	This is btnSha512Hash, a member of class FrmAnaSayfa.
	btnSifreUygunMu ( <a href="#">see page 36</a> )	This is btnSifreUygunMu, a member of class FrmAnaSayfa.
	btnTarihAraligi ( <a href="#">see page 36</a> )	This is btnTarihAraligi, a member of class FrmAnaSayfa.

	btnTcKimlikDogrula ( see page 36)	This is btnTcKimlikDogrula, a member of class FrmAnaSayfa.
	btnTelefonNoDuzenle ( see page 37)	This is btnTelefonNoDuzenle, a member of class FrmAnaSayfa.
	btnTurkceKarakterDuzelt ( see page 37)	This is btnTurkceKarakterDuzelt, a member of class FrmAnaSayfa.
	btnYasHesapla ( see page 37)	This is btnYasHesapla, a member of class FrmAnaSayfa.
	btnYasHesaplaMetinsel ( see page 37)	This is btnYasHesaplaMetinsel, a member of class FrmAnaSayfa.
	cmbIslemTuru ( see page 37)	This is cmbIslemTuru, a member of class FrmAnaSayfa.
	components ( see page 37)	Required designer variable.
	dtpBaslangicTarihi ( see page 37)	This is dtpBaslangicTarihi, a member of class FrmAnaSayfa.
	dtpBitisTarihi ( see page 38)	This is dtpBitisTarihi, a member of class FrmAnaSayfa.
	label1 ( see page 38)	This is label1, a member of class FrmAnaSayfa.
	label10 ( see page 38)	This is label10, a member of class FrmAnaSayfa.
	label11 ( see page 38)	This is label11, a member of class FrmAnaSayfa.
	label12 ( see page 38)	This is label12, a member of class FrmAnaSayfa.
	label13 ( see page 38)	This is label13, a member of class FrmAnaSayfa.
	label14 ( see page 38)	This is label14, a member of class FrmAnaSayfa.
	label15 ( see page 39)	This is label15, a member of class FrmAnaSayfa.
	label16 ( see page 39)	This is label16, a member of class FrmAnaSayfa.
	label17 ( see page 39)	This is label17, a member of class FrmAnaSayfa.
	label2 ( see page 39)	This is label2, a member of class FrmAnaSayfa.
	label3 ( see page 39)	This is label3, a member of class FrmAnaSayfa.
	label4 ( see page 39)	This is label4, a member of class FrmAnaSayfa.
	label5 ( see page 40)	This is label5, a member of class FrmAnaSayfa.
	label6 ( see page 40)	This is label6, a member of class FrmAnaSayfa.
	label7 ( see page 40)	This is label7, a member of class FrmAnaSayfa.
	label8 ( see page 40)	This is label8, a member of class FrmAnaSayfa.
	label9 ( see page 40)	This is label9, a member of class FrmAnaSayfa.
	lbKisiSonuc ( see page 40)	This is lbKisiSonuc, a member of class FrmAnaSayfa.
	lbMetinSonuc ( see page 40)	This is lbMetinSonuc, a member of class FrmAnaSayfa.
	lbSayiSonuc ( see page 41)	This is lbSayiSonuc, a member of class FrmAnaSayfa.
	lbSifreSonuc ( see page 41)	This is lbSifreSonuc, a member of class FrmAnaSayfa.
	lbSonuc ( see page 41)	This is lbSonuc, a member of class FrmAnaSayfa.
	pnlKisilslemleri ( see page 41)	This is pnlKisilslemleri, a member of class FrmAnaSayfa.
	pnlMetinIslemleri ( see page 41)	This is pnlMetinIslemleri, a member of class FrmAnaSayfa.
	pnlSayiIslemleri ( see page 41)	This is pnlSayiIslemleri, a member of class FrmAnaSayfa.
	pnlSifreIslemleri ( see page 42)	This is pnlSifreIslemleri, a member of class FrmAnaSayfa.
	pnlTarihIslemleri ( see page 42)	This is pnlTarihIslemleri, a member of class FrmAnaSayfa.
	txtKisiMetni ( see page 42)	This is txtKisiMetni, a member of class FrmAnaSayfa.
	txtMetin ( see page 42)	This is txtMetin, a member of class FrmAnaSayfa.
	txtSayi ( see page 42)	This is txtSayi, a member of class FrmAnaSayfa.
	txtSifrelenecekMetin ( see page 42)	This is txtSifrelenecekMetin, a member of class FrmAnaSayfa.

**FrmAnaSayfa Methods**

	Name	Description
	cmbIslemTuru_SelectedIndexChanged (see page 42)	Islem Türü Combobox'ında seçilen degere göre Ekrandaki panellerden ilgili olan haricinde hepsinin erişilemez hale getirilmesi Eger bir sey seçilmemisse, tamamının erişilemez hale getirilmesi için Gerekli işlemlerin gerçekleştirildiği metottur.
	Dispose (see page 51)	Clean up any resources being used.
	FrmAnaSayfa_Load (see page 51)	Form İlk Yüklendiği zaman ekrandaki bütün panellerin erişilemez hale getirilmesi sağlanıyor Bu metotta, çünkü hangi panelin aktifleşeceğinin Combobox üzerinden seçilmesi lazım
	InitializeComponent (see page 52)	Required method for Designer support - do not modify the contents of this method with the code editor.

**1.1.3.1.1 FrmAnaSayfa.FrmAnaSayfa Constructor****C#**

```
public FrmAnaSayfa();
```

**Description**

This is FrmAnaSayfa, a member of class FrmAnaSayfa.

**Body Source**

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

**1.1.3.1.2 FrmAnaSayfa Fields****1.1.3.1.2.1 FrmAnaSayfa.btnAyFarki Field****C#**

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

**Description**

This is btnAyFarki, a member of class FrmAnaSayfa.

**1.1.3.1.2.2 FrmAnaSayfa.btnEPostaDogrula Field****C#**

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

**Description**

This is btnEPostaDogrula, a member of class FrmAnaSayfa.

**1.1.3.1.2.3 FrmAnaSayfa.btnIbanDogrula Field****C#**

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

**Description**

This is btnIbanDogrula, a member of class FrmAnaSayfa.

#### 1.1.3.1.2.4 FrmAnaSayfa.btnIlkHarfiBuyukYap Field

**C#**

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

##### Description

This is btnIlkHarfiBuyukYap, a member of class FrmAnaSayfa.

#### 1.1.3.1.2.5 FrmAnaSayfa.btnIlkHarfleriBuyukYap Field

**C#**

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

##### Description

This is btnIlkHarfleriBuyukYap, a member of class FrmAnaSayfa.

#### 1.1.3.1.2.6 FrmAnaSayfa.btnMd5Hash Field

**C#**

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

##### Description

This is btnMd5Hash, a member of class FrmAnaSayfa.

#### 1.1.3.1.2.7 FrmAnaSayfa.btnMetinSifrele Field

**C#**

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

##### Description

This is btnMetinSifrele, a member of class FrmAnaSayfa.

#### 1.1.3.1.2.8 FrmAnaSayfa.btnMetneDonustur Field

**C#**

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

##### Description

This is btnMetneDonustur, a member of class FrmAnaSayfa.

#### 1.1.3.1.2.9 FrmAnaSayfa.btnOnayKoduOlustur Field

**C#**

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

##### Description

This is btnOnayKoduOlustur, a member of class FrmAnaSayfa.

#### 1.1.3.1.2.10 FrmAnaSayfa.btnRastgeleSifreUret Field

**C#**

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

**Description**

This is btnRastgeleSifreUret, a member of class FrmAnaSayfa.

**1.1.3.1.2.11 FrmAnaSayfa.btnSayisalMi Field****C#**

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

**Description**

This is btnSayisalMi, a member of class FrmAnaSayfa.

**1.1.3.1.2.12 FrmAnaSayfa.btnSha256Hash Field****C#**

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

**Description**

This is btnSha256Hash, a member of class FrmAnaSayfa.

**1.1.3.1.2.13 FrmAnaSayfa.btnSha512Hash Field****C#**

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

**Description**

This is btnSha512Hash, a member of class FrmAnaSayfa.

**1.1.3.1.2.14 FrmAnaSayfa.btnSifreUygunMu Field****C#**

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

**Description**

This is btnSifreUygunMu, a member of class FrmAnaSayfa.

**1.1.3.1.2.15 FrmAnaSayfa.btnTarihAraligi Field****C#**

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

**Description**

This is btnTarihAraligi, a member of class FrmAnaSayfa.

**1.1.3.1.2.16 FrmAnaSayfa.btnTcKimlikDogrula Field****C#**

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

**Description**

This is btnTcKimlikDogrula, a member of class FrmAnaSayfa.

### 1.1.3.1.2.17 FrmAnaSayfa.btnTelefonNoDuzenle Field

C#

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

#### Description

This is btnTelefonNoDuzenle, a member of class FrmAnaSayfa.

### 1.1.3.1.2.18 FrmAnaSayfa.btnTurkceKarakterDuzelt Field

C#

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

#### Description

This is btnTurkceKarakterDuzelt, a member of class FrmAnaSayfa.

### 1.1.3.1.2.19 FrmAnaSayfa.btnYasHesapla Field

C#

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

#### Description

This is btnYasHesapla, a member of class FrmAnaSayfa.

### 1.1.3.1.2.20 FrmAnaSayfa.btnYasHesaplaMetinsel Field

C#

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

#### Description

This is btnYasHesaplaMetinsel, a member of class FrmAnaSayfa.

### 1.1.3.1.2.21 FrmAnaSayfa.cmbIslemTuru Field

C#

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

#### Description

This is cmbIslemTuru, a member of class FrmAnaSayfa.

### 1.1.3.1.2.22 FrmAnaSayfa.components Field

Required designer variable.

C#

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

### 1.1.3.1.2.23 FrmAnaSayfa.dtpBaslangicTarihi Field

C#

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

#### Description

This is dtpBaslangicTarihi, a member of class FrmAnaSayfa.



#### 1.1.3.1.2.24 FrmAnaSayfa.dtpBitisTarihi Field

**C#**

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

**Description**

This is dtpBitisTarihi, a member of class FrmAnaSayfa.

#### 1.1.3.1.2.25 FrmAnaSayfa.label1 Field

**C#**

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

**Description**

This is label1, a member of class FrmAnaSayfa.

#### 1.1.3.1.2.26 FrmAnaSayfa.label10 Field

**C#**

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

**Description**

This is label10, a member of class FrmAnaSayfa.

#### 1.1.3.1.2.27 FrmAnaSayfa.label11 Field

**C#**

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

**Description**

This is label11, a member of class FrmAnaSayfa.

#### 1.1.3.1.2.28 FrmAnaSayfa.label12 Field

**C#**

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

**Description**

This is label12, a member of class FrmAnaSayfa.

#### 1.1.3.1.2.29 FrmAnaSayfa.label13 Field

**C#**

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

**Description**

This is label13, a member of class FrmAnaSayfa.

#### 1.1.3.1.2.30 FrmAnaSayfa.label14 Field

**C#**

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

**Description**

This is label14, a member of class FrmAnaSayfa.

**1.1.3.1.2.31 FrmAnaSayfa.label15 Field****C#**

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

**Description**

This is label15, a member of class FrmAnaSayfa.

**1.1.3.1.2.32 FrmAnaSayfa.label16 Field****C#**

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

**Description**

This is label16, a member of class FrmAnaSayfa.

**1.1.3.1.2.33 FrmAnaSayfa.label17 Field****C#**

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

**Description**

This is label17, a member of class FrmAnaSayfa.

**1.1.3.1.2.34 FrmAnaSayfa.label2 Field****C#**

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

**Description**

This is label2, a member of class FrmAnaSayfa.

**1.1.3.1.2.35 FrmAnaSayfa.label3 Field****C#**

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

**Description**

This is label3, a member of class FrmAnaSayfa.

**1.1.3.1.2.36 FrmAnaSayfa.label4 Field****C#**

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

**Description**

This is label4, a member of class FrmAnaSayfa.

#### 1.1.3.1.2.37 FrmAnaSayfa.label5 Field

**C#**

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

**Description**

This is label5, a member of class FrmAnaSayfa.

#### 1.1.3.1.2.38 FrmAnaSayfa.label6 Field

**C#**

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

**Description**

This is label6, a member of class FrmAnaSayfa.

#### 1.1.3.1.2.39 FrmAnaSayfa.label7 Field

**C#**

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

**Description**

This is label7, a member of class FrmAnaSayfa.

#### 1.1.3.1.2.40 FrmAnaSayfa.label8 Field

**C#**

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

**Description**

This is label8, a member of class FrmAnaSayfa.

#### 1.1.3.1.2.41 FrmAnaSayfa.label9 Field

**C#**

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

**Description**

This is label9, a member of class FrmAnaSayfa.

#### 1.1.3.1.2.42 FrmAnaSayfa.lbKisiSonuc Field

**C#**

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

**Description**

This is lbKisiSonuc, a member of class FrmAnaSayfa.

#### 1.1.3.1.2.43 FrmAnaSayfa.lbMetinSonuc Field

**C#**

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

**Description**

This is lbMetinSonuc, a member of class FrmAnaSayfa.

**1.1.3.1.2.44 FrmAnaSayfa.lbSayiSonuc Field****C#**

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

**Description**

This is lbSayiSonuc, a member of class FrmAnaSayfa.

**1.1.3.1.2.45 FrmAnaSayfa.lbSifreSonuc Field****C#**

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

**Description**

This is lbSifreSonuc, a member of class FrmAnaSayfa.

**1.1.3.1.2.46 FrmAnaSayfa.lbSonuc Field****C#**

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

**Description**

This is lbSonuc, a member of class FrmAnaSayfa.

**1.1.3.1.2.47 FrmAnaSayfa.pnlKisilslemleri Field****C#**

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

**Description**

This is pnlKisilslemleri, a member of class FrmAnaSayfa.

**1.1.3.1.2.48 FrmAnaSayfa.pnlMetinIslemleri Field****C#**

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

**Description**

This is pnlMetinIslemleri, a member of class FrmAnaSayfa.

**1.1.3.1.2.49 FrmAnaSayfa.pnlSayiIslemleri Field****C#**

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

**Description**

This is pnlSayiIslemleri, a member of class FrmAnaSayfa.

### 1.1.3.1.2.50 FrmAnaSayfa.pnlSifreIslemleri Field

C#

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

#### Description

This is pnlSifreIslemleri, a member of class FrmAnaSayfa.

### 1.1.3.1.2.51 FrmAnaSayfa.pnlTarihIslemleri Field

C#

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

#### Description

This is pnlTarihIslemleri, a member of class FrmAnaSayfa.

### 1.1.3.1.2.52 FrmAnaSayfa.txtKisiMetni Field

C#

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

#### Description

This is txtKisiMetni, a member of class FrmAnaSayfa.

### 1.1.3.1.2.53 FrmAnaSayfa.txtMetin Field

C#

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

#### Description

This is txtMetin, a member of class FrmAnaSayfa.

### 1.1.3.1.2.54 FrmAnaSayfa.txtSayi Field

C#

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

#### Description

This is txtSayi, a member of class FrmAnaSayfa.

### 1.1.3.1.2.55 FrmAnaSayfa.txtSifrelenecekMetin Field

C#

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

#### Description

This is txtSifrelenecekMetin, a member of class FrmAnaSayfa.

## 1.1.3.1.3 FrmAnaSayfa Methods

### 1.1.3.1.3.1 FrmAnaSayfa.cmbIslemTuru\_SelectedIndexChanged Method

Islem Türü Combobox'ında seçilen degere göre Ekrandaki panellerden ilgili olan haricinde hepsinin erişilemez hale

getirilmesi Eger bir sey seçilmemisse, tamamının erisilemez hale getirilmesi için Gerekli islemlerin gerçekleştirildiği metottur.

**C#**

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

#### Parameters

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

#### Body Source

```
1: private void cmbIslemTuru_SelectedIndexChanged(object sender, System.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:                 pnlMetinIslemleri.Enabled = true;
10:                pnlTarihIslemleri.Enabled = false;
11:                pnlSifreIslemleri.Enabled = false;
12:                pnlSayiIslemleri.Enabled = false;
13:                pnlKisiIslemleri.Enabled = false;
14:                break;
15:            }
16:            // Tarih Islemleri
17:            case 1:
18:            {
19:                pnlMetinIslemleri.Enabled = false;
20:                pnlTarihIslemleri.Enabled = true;
21:                pnlSifreIslemleri.Enabled = false;
22:                pnlSayiIslemleri.Enabled = false;
23:                pnlKisiIslemleri.Enabled = false;
24:                break;
25:            }
26:            // Sayi Islemleri
27:            case 2:
28:            {
29:                pnlMetinIslemleri.Enabled = false;
30:                pnlTarihIslemleri.Enabled = false;
31:                pnlSifreIslemleri.Enabled = false;
32:                pnlSayiIslemleri.Enabled = true;
33:                pnlKisiIslemleri.Enabled = false;
34:                break;
35:            }
36:            // Sifre Islemleri
37:            case 3:
38:            {
39:                pnlMetinIslemleri.Enabled = false;
40:                pnlTarihIslemleri.Enabled = false;
41:                pnlSifreIslemleri.Enabled = true;
42:                pnlSayiIslemleri.Enabled = false;
43:                pnlKisiIslemleri.Enabled = false;
44:                break;
45:            }
46:            // Kisi Islemleri
47:            case 4:
48:            {
49:                pnlMetinIslemleri.Enabled = false;
50:                pnlTarihIslemleri.Enabled = false;
51:                pnlSifreIslemleri.Enabled = false;
52:                pnlSayiIslemleri.Enabled = false;
53:                pnlKisiIslemleri.Enabled = true;
54:                break;
55:            }
56:        }
57:    }
```

```

56:                // Bunlar haricinde herhangi bir sey
57:                default:
58:                {
59:                    pnlMetinIslemleri.Enabled = false;
60:                    pnlTarihIslemleri.Enabled = false;
61:                    pnlSifreIslemleri.Enabled = false;
62:                    pnlSayiIslemleri.Enabled = false;
63:                    pnlKisiIslemleri.Enabled = false;
64:                    break;
65:                }
66:            }
67:        #endregion
68:    }
69:
70:    /// <summary>
71:    /// This method is used to measure the length of the item in listbox and make
it multi line
72:    /// without using scrollbar and let the user see more efficiently
73:    /// </summary>
74:    /// <param name="sender">The sender info (For example Main Form)</param>
75:    /// <param name="e">Event Arguments</param>
76:    private static void listBox_MeasureItem(object sender, MeasureItemEventArgs e)
77:    {
78:        if (!(sender is ListBox listBox)) return;
79:        e.ItemHeight = (int) e.Graphics
80:            .MeasureString(listBox.Items[e.Index].ToString(), listBox.Font,
listBox.Width).Height;
81:    }
82:
83:    /// <summary>
84:    /// This method is used to measure the length of the item in listbox and make
it multi line
85:    /// without using scrollbar and let the user see more efficiently
86:    /// </summary>
87:    /// <param name="sender">The sender info (For example Main Form)</param>
88:    /// <param name="e">Event Arguments</param>
89:    private static void listBox_DrawItem(object sender, DrawItemEventArgs e)
90:    {
91:        if (!(sender is ListBox listBox)) return;
92:        e.DrawBackground();
93:        e.DrawFocusRectangle();
94:        e.Graphics.DrawString(listBox.Items[e.Index].ToString(), e.Font, new
SolidBrush(e.ForeColor), e.Bounds);
95:    }
96:
97:    #endregion
98:
99:    #region Tarih Islemleri
100:
101:    /// <summary>
102:    /// Paneldeki Baslangiç Tarihi seçili ve tarih girilmissse,
103:    /// o tarihten bugüne kaç yas oldugunun hesaplandigi metottur.
104:    /// Ilgili butona tiklandigi zaman çalisir
105:    /// </summary>
106:    /// <param name="sender">The sender info (For example Main Form)</param>
107:    /// <param name="e">Event Arguments</param>
108:    private void btnYasHesapla_Click(object sender, System.EventArgs e)
109:    {
110:        if (!dtpBaslangicTarihi.Checked)
111:            MessageBox.Show("Baslangiç Tarihi girilmedigi için bu islem
gerçeklestirilemez");
112:        else
113:        {
114:            lbSonuc.Items.Clear();
115:            var sb = new StringBuilder();
116:            sb.Append("Dogum tarihi seçilen kisi ");
117:            sb.Append(TarihIslemleri.YasHesapla(dtpBaslangicTarihi.Value));
118:            sb.Append(" yasinin içindedir");
119:            lbSonuc.Items.Add(sb.ToString());

```

```

120:         }
121:     }
122:
123:     /// <summary>
124:     /// Paneldeki Baslangiç Tarihi seçili ve tarih girilirse,
125:     /// o tarihten bugüne kaç yas oldugunun yıl, ay ve gün
126:     /// olarak metinsel bir sekilde hesaplandigi metottur.
127:     /// Ilgili butona tiklandigi zaman çalışır
128:     /// </summary>
129:     /// <param name="sender">The sender info (For example Main Form)</param>
130:     /// <param name="e">Event Arguments</param>
131:     private void btnYasHesaplaMetinsel_Click(object sender, System.EventArgs e)
132:     {
133:         if (!dtpBaslangicTarihi.Checked)
134:             MessageBox.Show("Baslangiç Tarihi girilmediği için bu işlem
gerçekleştiremez");
135:         else
136:         {
137:             lbSonuc.Items.Clear();
138:             var sb = new StringBuilder();
139:             sb.Append(TarihIslemleri.YasHesaplaMetinsel(dtpBaslangicTarihi.Value));
140:             lbSonuc.Items.Add(sb.ToString());
141:         }
142:     }
143:
144:     /// <summary>
145:     /// Paneldeki Baslangiç Tarihi ve Bitis Tarihi alanlari seçili ve tarih
girilirse,
146:     /// Söz konusu tarihler arasında kaç yıl, kaç ay ve gün oldugunun bilgisinin
hazirlandigi metottur.
147:     /// Ilgili butona tiklandigi zaman çalışır
148:     /// </summary>
149:     /// <param name="sender">The sender info (For example Main Form)</param>
150:     /// <param name="e">Event Arguments</param>
151:     private void btnTarihAraligi_Click(object sender, System.EventArgs e)
152:     {
153:         if (!dtpBaslangicTarihi.Checked || !dtpBitisTarihi.Checked)
154:             MessageBox.Show("Baslangiç Tarihi veya Bitis Tarihi girilmediği için
bu işlem gerçekleştiremez");
155:         else if (dtpBaslangicTarihi.Value.Date > dtpBitisTarihi.Value.Date)
156:             MessageBox.Show("Baslangiç Tarihi Bitis Tarihinden büyük olamaz");
157:         else
158:         {
159:             lbSonuc.Items.Clear();
160:             var sb = new StringBuilder();
161:             sb.Append("İki tarih arası: ");
162:             sb.Append(TarihIslemleri.TarihAraligiHesapla(dtpBaslangicTarihi.Value.Date,
dtpBitisTarihi.Value.Date));
163:             lbSonuc.Items.Add(sb.ToString());
164:         }
165:     }
166:
167:     /// <summary>
168:     /// Paneldeki Baslangiç Tarihi ve Bitis Tarihi alanlari seçili ve tarih
girilirse,
169:     /// Söz konusu tarihler arasında kaç ay oldugunun bilgisinin hazirlandigi
metottur.
170:     /// Ilgili butona tiklandigi zaman çalışır
171:     /// </summary>
172:     /// <param name="sender">The sender info (For example Main Form)</param>
173:     /// <param name="e">Event Arguments</param>
174:     private void btnAyFarki_Click(object sender, System.EventArgs e)
175:     {
176:         if (!dtpBaslangicTarihi.Checked || !dtpBitisTarihi.Checked)
177:             MessageBox.Show("Baslangiç Tarihi veya Bitis Tarihi girilmediği için
bu işlem gerçekleştiremez");
178:         else if (dtpBaslangicTarihi.Value.Date > dtpBitisTarihi.Value.Date)
179:             MessageBox.Show("Baslangiç Tarihi Bitis Tarihinden büyük olamaz");

```



```

180:         else
181:         {
182:             lbSonuc.Items.Clear();
183:             var sb = new StringBuilder();
184:             sb.Append("Ay Farki: ");
185:             sb.Append(TarihIslemleri.AyFarki(dtpBaslangicTarihi.Value.Date,
dtpBitisTarihi.Value.Date));
186:             lbSonuc.Items.Add(sb.ToString());
187:         }
188:     }
189: #endregion
190:
191: #region Sifre Islemleri
192: /// <summary>
193: /// Paneldeki metin kutusu dolu ise
194: /// Girilen metnin MD5 algoritmasi ile sifrelenmis halini
195: /// Listbox üzerine yazdiran metottur.
196: /// </summary>
197: /// <param name="sender">The sender info (For example Main Form)</param>
198: /// <param name="e">Event Arguments</param>
199: private void btnMd5Hash_Click(object sender, System.EventArgs e)
200: {
201:     if (string.IsNullOrEmpty(txtSifrelenecekMetin.Text.Trim()))
202:         MessageBox.Show("Sifrelenecek metin girilmedigi için bu işlem
gerçeklestirilemez");
203:     else
204:     {
205:         lbSifreSonuc.Items.Clear();
206:         var sb = new StringBuilder();
207:         sb.Append("Girilen Metnin MD5 Karsiligi: ");
208:         sb.Append(SifrelemeIslemleri.HesaplaMD5(txtSifrelenecekMetin.Text.Trim()));
209:         lbSifreSonuc.Items.Add(sb.ToString());
210:     }
211: }
212:
213: /// <summary>
214: /// Paneldeki metin kutusu dolu ise
215: /// Girilen metnin SHA 256 algoritmasi ile sifrelenmis halini
216: /// Listbox üzerine yazdiran metottur.
217: /// </summary>
218: /// <param name="sender">The sender info (For example Main Form)</param>
219: /// <param name="e">Event Arguments</param>
220: private void btnSha256Hash_Click(object sender, System.EventArgs e)
221: {
222:     if (string.IsNullOrEmpty(txtSifrelenecekMetin.Text.Trim()))
223:         MessageBox.Show("Sifrelenecek metin girilmedigi için bu işlem
gerçeklestirilemez");
224:     else
225:     {
226:         lbSifreSonuc.Items.Clear();
227:         var sb = new StringBuilder();
228:         sb.Append("Girilen Metnin SHA 256 Karsiligi: ");
229:         sb.Append(SifrelemeIslemleri.HesaplaSHA256(txtSifrelenecekMetin.Text.Trim()));
230:         lbSifreSonuc.Items.Add(sb.ToString());
231:     }
232: }
233:
234: /// <summary>
235: /// Paneldeki metin kutusu dolu ise
236: /// Girilen metnin SHA 512 algoritmasi ile sifrelenmis halini
237: /// Listbox üzerine yazdiran metottur.
238: /// </summary>
239: /// <param name="sender">The sender info (For example Main Form)</param>
240: /// <param name="e">Event Arguments</param>
241: private void btnSha512Hash_Click(object sender, System.EventArgs e)
242: {
243:     if (string.IsNullOrEmpty(txtSifrelenecekMetin.Text.Trim()))

```

```

244:                MessageBox.Show("Sifrelenecek metin girilmediği için bu işlem
gerçekleştirilemez");
245:            else
246:            {
247:                lbSifreSonuc.Items.Clear();
248:                var sb = new StringBuilder();
249:                sb.Append("Girilen Metnin SHA 512 Karsiligi: ");
250:
sb.Append(SifrelemeIslemleri.HesaplaSHA512(txtSifrelenecekMetin.Text.Trim()));
251:                lbSifreSonuc.Items.Add(sb.ToString());
252:            }
253:        }
254:
255:        /// <summary>
256:        /// ilgili butona tiklandiginda rastgele sifre üretilmesini saglayan metottur
257:        /// </summary>
258:        /// <param name="sender">The sender info (For example Main Form)</param>
259:        /// <param name="e">Event Arguments</param>
260:        private void btnRastgeleSifreUret_Click(object sender, System.EventArgs e)
261:        {
262:            lbSifreSonuc.Items.Clear();
263:            var sb = new StringBuilder();
264:            sb.Append("Rastgele Üretilen Sifre: ");
265:            sb.Append(SifrelemeIslemleri.RastgeleSifreUret());
266:            lbSifreSonuc.Items.Add(sb.ToString());
267:        }
268:
269:        /// <summary>
270:        /// Paneldeki metin kutusu dolu ise
271:        /// Burada yazılan metnin bir sifre olarak belirli kriterlere uygun olup
olmadiginin
272:        /// Gösterilmesini saglayan bir metottur
273:        /// </summary>
274:        /// <param name="sender">The sender info (For example Main Form)</param>
275:        /// <param name="e">Event Arguments</param>
276:        private void btnSifreUygunMu_Click(object sender, System.EventArgs e)
277:        {
278:            if (string.IsNullOrEmpty(txtSifrelenecekMetin.Text.Trim()))
279:                MessageBox.Show("Sifre metni girilmediği için bu işlem
gerçekleştirilemez");
280:            else
281:            {
282:                lbSifreSonuc.Items.Clear();
283:                var sb = new StringBuilder();
284:                sb.Append("Girilen Sifrenin Uygunluk Durumu: ");
285:
sb.Append(SifrelemeIslemleri.SifreUygunMu(txtSifrelenecekMetin.Text.Trim()));
286:                lbSifreSonuc.Items.Add(sb.ToString());
287:            }
288:        }
289:    #endregion
290:
291:    #region Sayi Islemleri
292:    /// <summary>
293:    /// Paneldeki metin kutusuna girilen metnin sayisal bir deger olup olmadigini
294:    /// Kontrol eden ve sonucu gösteren metottur.
295:    /// </summary>
296:    /// <param name="sender">The sender info (For example Main Form)</param>
297:    /// <param name="e">Event Arguments</param>
298:    private void btnSayisalMi_Click(object sender, System.EventArgs e)
299:    {
300:        if (string.IsNullOrEmpty(txtSayi.Text.Trim()))
301:            MessageBox.Show("Metin girilmediği için bu işlem gerçekleştirilemez");
302:        else
303:        {
304:            lbSayiSonuc.Items.Clear();
305:            var sb = new StringBuilder();
306:            sb.Append("Girilen Metin Sayi Mi?: ");
307:            sb.Append(SayiIslemleri.SayisalMi(txtSayi.Text.Trim()) ? "Evet" :

```

```

"Hayir");
308:         lbSayiSonuc.Items.Add(sb.ToString());
309:     }
310: }
311:
312: /// <summary>
313: /// Paneldeki metin kutusuna girilen para birimi cinsinden rakamin
314: /// Yaziya çevrilmesini saglayan ve sonucu listbox üzerinde gösteren bir
metottur.
315: /// </summary>
316: /// <param name="sender">The sender info (For example Main Form)</param>
317: /// <param name="e">Event Arguments</param>
318: private void btnMetneDonustur_Click(object sender, System.EventArgs e)
319: {
320:     if (string.IsNullOrEmpty(txtSayi.Text.Trim()))
321:         MessageBox.Show("Metin girilmedigi için bu islem gerçekleştirilemez");
322:     else if (!SayiIslemleri.SayisalMi(txtSayi.Text.Trim().Replace(".", "")))
323:         MessageBox.Show("Girilen metin sayisal bir deger olmadigi için bu
islem gerçekleştirilemez");
324:     else
325:     {
326:         var metinSadeceSayiKismi = txtSayi.Text.Trim();
327:         var sayi = SayiIslemleri.SayiHalineCevir(metinSadeceSayiKismi);
328:         lbSayiSonuc.Items.Clear();
329:         var sb = new StringBuilder();
330:         sb.Append("Girilen Sayinin Metin Hali (Parasal): ");
331:         sb.Append(SayiIslemleri.SayiyiYaziyaCevir(sayi));
332:         lbSayiSonuc.Items.Add(sb.ToString());
333:     }
334: }
335:
336: /// <summary>
337: /// Rastgele olacak sekilde 6 haneli kod olusturmayi saglayan bir metottur.
338: /// </summary>
339: /// <param name="sender">The sender info (For example Main Form)</param>
340: /// <param name="e">Event Arguments</param>
341: private void btnOnayKoduOlustur_Click(object sender, System.EventArgs e)
342: {
343:     lbSayiSonuc.Items.Clear();
344:     var sb = new StringBuilder();
345:     sb.Append("Olusturulan 6 haneli onay kodu: ");
346:     sb.Append(SayiIslemleri.AltiHaneliOnayKoduOlustur());
347:     lbSayiSonuc.Items.Add(sb.ToString());
348: }
349: #endregion
350:
351: #region Kisi Islemleri
352: /// <summary>
353: /// Paneldeki metin kutusuna girilen TC Kimlik numarasinin
354: /// Standart TC Kimlik No algoritmasina uygun olup olmadigini kontrol eden bir
metottur.
355: /// </summary>
356: /// <param name="sender">The sender info (For example Main Form)</param>
357: /// <param name="e">Event Arguments</param>
358: private void btnTcKimlikDogrula_Click(object sender, System.EventArgs e)
359: {
360:     if (string.IsNullOrEmpty(txtKisiMetni.Text.Trim()))
361:         MessageBox.Show("Metin girilmedigi için bu islem gerçekleştirilemez");
362:     else if (!SayiIslemleri.SayisalMi(txtKisiMetni.Text.Trim()))
363:         MessageBox.Show("Girilen metin sayisal bir deger olmadigi için bu
islem gerçekleştirilemez");
364:     else if (txtKisiMetni.Text.Trim().Length != 11)
365:         MessageBox.Show("Girilen metin 11 karakter olmadigi için bu islem
gerçekleştirilemez");
366:     else
367:     {
368:         lbKisiSonuc.Items.Clear();
369:         var sb = new StringBuilder();

```

```

370:                sb.Append("TC Kimlik No Dogru Mu?: ");
371:                sb.Append(KisiIslemleri.DogrulaTcKimlikNo(txtKisiMetni.Text.Trim()) ?
"Evet" : "Hayir");
372:                lbKisiSonuc.Items.Add(sb.ToString());
373:            }
374:        }
375:
376:        /// <summary>
377:        /// Paneldeki metin kutusuna girilen IBAN Numarasinin
378:        /// Standart IBAN algoritmasina uygun olup olmadigini kontrol eden bir
metottur.
379:        /// </summary>
380:        /// <param name="sender">The sender info (For example Main Form)</param>
381:        /// <param name="e">Event Arguments</param>
382:        private void btnIbanDogrula_Click(object sender, System.EventArgs e)
383:        {
384:            if (string.IsNullOrEmpty(txtKisiMetni.Text.Trim()))
385:                MessageBox.Show("Metin girilmedigi için bu islem gerçekleştirilemez");
386:            else if (txtKisiMetni.Text.Trim().Length != 26)
387:                MessageBox.Show("Girilen metin 26 karakter olmadigi için bu islem
gerçekleştirilemez. Karakter sayisi: " + txtKisiMetni.Text.Trim().Length);
388:            else
389:            {
390:                lbKisiSonuc.Items.Clear();
391:                var sb = new StringBuilder();
392:                sb.Append("IBAN Dogru Mu?: ");
393:                sb.Append(KisiIslemleri.DogrulaIBAN(txtKisiMetni.Text.Trim()) ? "Evet"
: "Hayir");
394:                lbKisiSonuc.Items.Add(sb.ToString());
395:            }
396:        }
397:
398:        /// <summary>
399:        /// Paneldeki metin kutusuna girilen metnin
400:        /// Geçerli bir e-posta adresi olup olmadigini kontrol eden bir metottur.
401:        /// </summary>
402:        /// <param name="sender">The sender info (For example Main Form)</param>
403:        /// <param name="e">Event Arguments</param>
404:        private void btnEPostaDogrula_Click(object sender, System.EventArgs e)
405:        {
406:            if (string.IsNullOrEmpty(txtKisiMetni.Text.Trim()))
407:                MessageBox.Show("Metin girilmedigi için bu islem gerçekleştirilemez");
408:            else
409:            {
410:                lbKisiSonuc.Items.Clear();
411:                var sb = new StringBuilder();
412:                sb.Append("E-Posta Dogru Mu?: ");
413:                sb.Append(KisiIslemleri.DogrulaEPosta(txtKisiMetni.Text.Trim()) ?
"Evet" : "Hayir");
414:                lbKisiSonuc.Items.Add(sb.ToString());
415:            }
416:        }
417:        #endregion
418:
419:        #region Metin Islemleri
420:        /// <summary>
421:        /// Paneldeki metin kutusuna girilen metnin
422:        /// Kelimelerinin ilk harflerini büyük yapmaya yarayan bir metottur.
423:        /// </summary>
424:        /// <param name="sender">The sender info (For example Main Form)</param>
425:        /// <param name="e">Event Arguments</param>
426:        private void btnIlkHarfleriBuyukYap_Click(object sender, System.EventArgs e)
427:        {
428:            if (string.IsNullOrEmpty(txtMetin.Text.Trim()))
429:                MessageBox.Show("Metin girilmedigi için bu islem gerçekleştirilemez");
430:            else
431:            {
432:                lbMetinSonuc.Items.Clear();
433:                var sb = new StringBuilder();

```

```

434:         sb.Append("Metnin Düzenlenmis Hali: ");
435:         sb.Append(txtMetin.Text.Trim().MetinIlkKarakterleriBuyukYap());
436:         lbMetinSonuc.Items.Add(sb.ToString());
437:     }
438: }
439:
440: /// <summary>
441: /// Paneldeki metin kutusuna girilen metnin
442: /// Sadece ilk harfini büyük yapmaya yarayan bir metottur.
443: /// </summary>
444: /// <param name="sender">The sender info (For example Main Form)</param>
445: /// <param name="e">Event Arguments</param>
446: private void btnIlkHarfiBuyukYap_Click(object sender, System.EventArgs e)
447: {
448:     if (string.IsNullOrEmpty(txtMetin.Text.Trim()))
449:         MessageBox.Show("Metin girilmedigi için bu islem gerçekleştirilemez");
450:     else
451:     {
452:         lbMetinSonuc.Items.Clear();
453:         var sb = new StringBuilder();
454:         sb.Append("Metnin Düzenlenmis Hali: ");
455:         sb.Append(MetinIslemleri.MetinIlkKarakterBuyukDigerleriKucukYap(txtMetin.Text.Trim()));
456:         lbMetinSonuc.Items.Add(sb.ToString());
457:     }
458: }
459:
460: /// <summary>
461: /// Paneldeki metin kutusuna girilen metnin
462: /// İçerisindeki Türkçe karakterleri Ingilizce hallerine çeviren bir metottur.
463: /// </summary>
464: /// <param name="sender">The sender info (For example Main Form)</param>
465: /// <param name="e">Event Arguments</param>
466: private void btnTurkceKarakterDuzelt_Click(object sender, System.EventArgs e)
467: {
468:     if (string.IsNullOrEmpty(txtMetin.Text.Trim()))
469:         MessageBox.Show("Metin girilmedigi için bu islem gerçekleştirilemez");
470:     else
471:     {
472:         lbMetinSonuc.Items.Clear();
473:         var sb = new StringBuilder();
474:         sb.Append("Metnin Düzenlenmis Hali: ");
475:         sb.Append(MetinIslemleri.TurkceKarakterleriDuzelt(txtMetin.Text.Trim()));
476:         lbMetinSonuc.Items.Add(sb.ToString());
477:     }
478: }
479:
480: /// <summary>
481: /// KVKK durumlarından dolayı TC Kimlik No ve Vergi No gibi alanların
482: /// Matbu çıktılarda gizlenmesi gibi bir durum söz konusu olduğundan
483: /// Girilen bu 10 veya 11 karakterlik metinleri sifrelemeye yarayan bir
484: metottur
485: /// </summary>
486: /// <param name="sender">The sender info (For example Main Form)</param>
487: /// <param name="e">Event Arguments</param>
488: private void btnMetinSifrele_Click(object sender, System.EventArgs e)
489: {
490:     if (string.IsNullOrEmpty(txtMetin.Text.Trim()))
491:         MessageBox.Show("Metin girilmedigi için bu islem gerçekleştirilemez");
492:     else if (!SayiIslemleri.SayisalMi(txtMetin.Text.Trim()))
493:         MessageBox.Show("Girilen metin sayisal bir deger olmadigi için bu
494: islem gerçekleştirilemez");
495:     else if (txtMetin.Text.Trim().Length != 11 && txtMetin.Text.Trim().Length
496: != 10)
497:         MessageBox.Show("Girilen metin 10 veya 11 karakter olmadigi için bu
498: islem gerçekleştirilemez");
499:     else
500:     {

```

```

497:         lbMetinSonuc.Items.Clear();
498:         var sb = new StringBuilder();
499:         sb.Append("TC Kimlik No veya Vergi No'nun Sifrelenmis Hali: ");
500:         sb.Append(MetinIslemleri.MetinSifrele(txtMetin.Text.Trim()));
501:         lbMetinSonuc.Items.Add(sb.ToString());
502:     }
503: }
504:
505: /// <summary>
506: /// Ekrandan çeşitli karakterler içeren telefon numaralari alindigi zaman
507: /// Bu telefon numarasini olmasi gerektigi hale çevirmeye yarayan bir metottur.
508: /// </summary>
509: /// <param name="sender">The sender info (For example Main Form)</param>
510: /// <param name="e">Event Arguments</param>
511: private void btnTelefonNoDuzenle_Click(object sender, System.EventArgs e)
512: {
513:     if (string.IsNullOrEmpty(txtMetin.Text.Trim()))
514:         MessageBox.Show("Metin girilmediği için bu işlem gerçekleştirilemez");
515:     else
516:     {
517:         lbMetinSonuc.Items.Clear();
518:         var sb = new StringBuilder();
519:         sb.Append("Telefon Numarasinin Düzenlenmis Hali: ");
520:         sb.Append(MetinIslemleri.TelefonNoDuzenle(txtMetin.Text.Trim()));
521:         lbMetinSonuc.Items.Add(sb.ToString());
522:     }
523: }
524: #endregion
525: }
526: }

```

### 1.1.3.1.3.2 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: }

```

### 1.1.3.1.3.3 FrmAnaSayfa.FrmAnaSayfa\_Load Method

Form İlk Yüklendiği zaman ekrandaki bütün panellerin erişilemez hale getirilmesi sağlanıyor Bu metotta, çünkü hangi panelin aktifleseyeceğinin Combobox üzerinden seçilmesi lazım

**C#**

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

**Parameters**

Parameters	Description
object sender	The sender info (For example Main Form)

System.EventArgs e

Event Arguments

**Body Source**

```

1: private void FrmAnaSayfa_Load(object sender, System.EventArgs e)
2: {
3:     cmbIslemTuru.DropDownStyle = ComboBoxStyle.DropDownList;
4:     #region Sayfa İlk Yüklendiğinde Tüm Panellerin Erisime Kapanması
5:     pnlTarihIslemleri.Enabled = false;
6:     pnlSifreIslemleri.Enabled = false;
7:     pnlSayiIslemleri.Enabled = false;
8:     pnlKisiIslemleri.Enabled = false;
9:     pnlMetinIslemleri.Enabled = false;
10:    #endregion
11:    #region Bütün Listboxların MultiLine Yapılması
12:    lbMetinSonuc.DrawMode = DrawMode.OwnerDrawVariable;
13:    lbMetinSonuc.MeasureItem += listBox_MeasureItem;
14:    lbMetinSonuc.DrawItem += listBox_DrawItem;
15:    lbSayiSonuc.DrawMode = DrawMode.OwnerDrawVariable;
16:    lbSayiSonuc.MeasureItem += listBox_MeasureItem;
17:    lbSayiSonuc.DrawItem += listBox_DrawItem;
18:    lbKisiSonuc.DrawMode = DrawMode.OwnerDrawVariable;
19:    lbKisiSonuc.MeasureItem += listBox_MeasureItem;
20:    lbKisiSonuc.DrawItem += listBox_DrawItem;
21:    lbSifreSonuc.DrawMode = DrawMode.OwnerDrawVariable;
22:    lbSifreSonuc.MeasureItem += listBox_MeasureItem;
23:    lbSifreSonuc.DrawItem += listBox_DrawItem;
24:    lbSonuc.DrawMode = DrawMode.OwnerDrawVariable;
25:    lbSonuc.MeasureItem += listBox_MeasureItem;
26:    lbSonuc.DrawItem += listBox_DrawItem;
27:    #endregion
28: }

```

**1.1.3.1.3.4 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.txtSifrelenenecekMetin = 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.pnlTarihIslemleri.SuspendLayout();
58:     this.pnlSifreIslemleri.SuspendLayout();
59:     this.pnlSayiIslemleri.SuspendLayout();
60:     this.pnlKisiIslemleri.SuspendLayout();
61:     this.pnlMetinIslemleri.SuspendLayout();
62:     this.SuspendLayout();
63:     //
64:     // cmbIslemTuru
65:     //
66:     this.cmbIslemTuru.FormattingEnabled = true;
67:     this.cmbIslemTuru.Items.AddRange(new object[] { "Metin Islemleri", "Tarih
Islemleri", "Sayi Islemleri", "Sifre Islemleri", "Kisi Islemleri"});
68:     this.cmbIslemTuru.Location = new System.Drawing.Point(132, 13);
69:     this.cmbIslemTuru.Name = "cmbIslemTuru";
70:     this.cmbIslemTuru.Size = new System.Drawing.Size(200, 21);
71:     this.cmbIslemTuru.TabIndex = 0;
72:     this.cmbIslemTuru.SelectedIndexChanged += new
System.EventHandler(this.cmbIslemTuru_SelectedIndexChanged);
73:     //
74:     // label1
75:     //
76:     this.label1.AutoSize = true;
77:     this.label1.Location = new System.Drawing.Point(12, 16);
78:     this.label1.Name = "label1";
79:     this.label1.Size = new System.Drawing.Size(93, 13);
80:     this.label1.TabIndex = 1;
81:     this.label1.Text = "Islem Türü Seçiniz";
82:     //
83:     // pnlTarihIslemleri
84:     //
85:     this.pnlTarihIslemleri.BorderStyle = System.Windows.Forms.BorderStyle.FixedSingle;
86:     this.pnlTarihIslemleri.Controls.Add(this.btnAyFarki);
87:     this.pnlTarihIslemleri.Controls.Add(this.label8);
88:     this.pnlTarihIslemleri.Controls.Add(this.btnTarihAraligi);
89:     this.pnlTarihIslemleri.Controls.Add(this.btnYasHesaplaMetinsel);
90:     this.pnlTarihIslemleri.Controls.Add(this.btnYasHesapla);
91:     this.pnlTarihIslemleri.Controls.Add(this.label4);
92:     this.pnlTarihIslemleri.Controls.Add(this.lbSonuc);
93:     this.pnlTarihIslemleri.Controls.Add(this.dtpBitisTarihi);

```



```

94:         this.pnlTarihIslemleri.Controls.Add(this.dtpBaslangicTarihi);
95:         this.pnlTarihIslemleri.Controls.Add(this.label3);
96:         this.pnlTarihIslemleri.Controls.Add(this.label2);
97:         this.pnlTarihIslemleri.Location = new System.Drawing.Point(15, 56);
98:         this.pnlTarihIslemleri.Name = "pnlTarihIslemleri";
99:         this.pnlTarihIslemleri.Size = new System.Drawing.Size(374, 181);
100:        this.pnlTarihIslemleri.TabIndex = 2;
101:        //
102:        // btnAyFarki
103:        //
104:        this.btnAyFarki.Location = new System.Drawing.Point(286, 76);
105:        this.btnAyFarki.Name = "btnAyFarki";
106:        this.btnAyFarki.Size = new System.Drawing.Size(79, 23);
107:        this.btnAyFarki.TabIndex = 12;
108:        this.btnAyFarki.Text = "Ay Farki";
109:        this.btnAyFarki.UseVisualStyleBackColor = true;
110:        this.btnAyFarki.Click += new System.EventHandler(this.btnAyFarki_Click);
111:        //
112:        // label8
113:        //
114:        this.label8.AutoSize = true;
115:        this.label8.Dock = System.Windows.Forms.DockStyle.Top;
116:        this.label8.Location = new System.Drawing.Point(0, 0);
117:        this.label8.Name = "label8";
118:        this.label8.Size = new System.Drawing.Size(71, 13);
119:        this.label8.TabIndex = 11;
120:        this.label8.Text = "Tarih Islemleri";
121:        //
122:        // btnTarihAraligi
123:        //
124:        this.btnTarihAraligi.Location = new System.Drawing.Point(204, 76);
125:        this.btnTarihAraligi.Name = "btnTarihAraligi";
126:        this.btnTarihAraligi.Size = new System.Drawing.Size(76, 23);
127:        this.btnTarihAraligi.TabIndex = 10;
128:        this.btnTarihAraligi.Text = "Tarih Araligi";
129:        this.btnTarihAraligi.UseVisualStyleBackColor = true;
130:        this.btnTarihAraligi.Click += new System.EventHandler(this.btnTarihAraligi_Click);
131:        //
132:        // btnYasHesaplaMetinsel
133:        //
134:        this.btnYasHesaplaMetinsel.Location = new System.Drawing.Point(84, 76);
135:        this.btnYasHesaplaMetinsel.Name = "btnYasHesaplaMetinsel";
136:        this.btnYasHesaplaMetinsel.Size = new System.Drawing.Size(113, 23);
137:        this.btnYasHesaplaMetinsel.TabIndex = 9;
138:        this.btnYasHesaplaMetinsel.Text = "Yas Hesapla (Metin)";
139:        this.btnYasHesaplaMetinsel.UseVisualStyleBackColor = true;
140:        this.btnYasHesaplaMetinsel.Click += new
System.EventHandler(this.btnYasHesaplaMetinsel_Click);
141:        //
142:        // btnYasHesapla
143:        //
144:        this.btnYasHesapla.Location = new System.Drawing.Point(3, 76);
145:        this.btnYasHesapla.Name = "btnYasHesapla";
146:        this.btnYasHesapla.Size = new System.Drawing.Size(75, 23);
147:        this.btnYasHesapla.TabIndex = 8;
148:        this.btnYasHesapla.Text = "Yas Hesapla";
149:        this.btnYasHesapla.UseVisualStyleBackColor = true;
150:        this.btnYasHesapla.Click += new System.EventHandler(this.btnYasHesapla_Click);
151:        //
152:        // label4
153:        //
154:        this.label4.AutoSize = true;
155:        this.label4.Location = new System.Drawing.Point(18, 109);
156:        this.label4.Name = "label4";
157:        this.label4.Size = new System.Drawing.Size(38, 13);
158:        this.label4.TabIndex = 7;
159:        this.label4.Text = "Sonuç";
160:        //
161:        // lbSonuc

```

```
162:    //
163:    this.lbSonuc.FormattingEnabled = true;
164:    this.lbSonuc.HorizontalScrollbar = true;
165:    this.lbSonuc.Location = new System.Drawing.Point(62, 105);
166:    this.lbSonuc.Name = "lbSonuc";
167:    this.lbSonuc.Size = new System.Drawing.Size(256, 69);
168:    this.lbSonuc.TabIndex = 6;
169:    //
170:    // dtpBitisTarihi
171:    //
172:    this.dtpBitisTarihi.Format = System.Windows.Forms.DateTimePickerFormat.Short;
173:    this.dtpBitisTarihi.Location = new System.Drawing.Point(117, 43);
174:    this.dtpBitisTarihi.Name = "dtpBitisTarihi";
175:    this.dtpBitisTarihi.ShowCheckBox = true;
176:    this.dtpBitisTarihi.Size = new System.Drawing.Size(200, 20);
177:    this.dtpBitisTarihi.TabIndex = 5;
178:    //
179:    // dtpBaslangicTarihi
180:    //
181:    this.dtpBaslangicTarihi.Format = System.Windows.Forms.DateTimePickerFormat.Short;
182:    this.dtpBaslangicTarihi.Location = new System.Drawing.Point(117, 17);
183:    this.dtpBaslangicTarihi.Name = "dtpBaslangicTarihi";
184:    this.dtpBaslangicTarihi.ShowCheckBox = true;
185:    this.dtpBaslangicTarihi.Size = new System.Drawing.Size(200, 20);
186:    this.dtpBaslangicTarihi.TabIndex = 4;
187:    //
188:    // label3
189:    //
190:    this.label3.AutoSize = true;
191:    this.label3.Location = new System.Drawing.Point(44, 47);
192:    this.label3.Name = "label3";
193:    this.label3.Size = new System.Drawing.Size(55, 13);
194:    this.label3.TabIndex = 3;
195:    this.label3.Text = "Bitis Tarihi";
196:    //
197:    // label2
198:    //
199:    this.label2.AutoSize = true;
200:    this.label2.Location = new System.Drawing.Point(17, 24);
201:    this.label2.Name = "label2";
202:    this.label2.Size = new System.Drawing.Size(82, 13);
203:    this.label2.TabIndex = 2;
204:    this.label2.Text = "Baslangiç Tarihi";
205:    //
206:    // pnlSifreIslemleri
207:    //
208:    this.pnlSifreIslemleri.BorderStyle = System.Windows.Forms.BorderStyle.FixedSingle;
209:    this.pnlSifreIslemleri.Controls.Add(this.label7);
210:    this.pnlSifreIslemleri.Controls.Add(this.btnSifreUygunMu);
211:    this.pnlSifreIslemleri.Controls.Add(this.btnRastgeleSifreUret);
212:    this.pnlSifreIslemleri.Controls.Add(this.btnSha512Hash);
213:    this.pnlSifreIslemleri.Controls.Add(this.btnSha256Hash);
214:    this.pnlSifreIslemleri.Controls.Add(this.label6);
215:    this.pnlSifreIslemleri.Controls.Add(this.lbSifreSonuc);
216:    this.pnlSifreIslemleri.Controls.Add(this.btnMd5Hash);
217:    this.pnlSifreIslemleri.Controls.Add(this.txtSifrelenecekMetin);
218:    this.pnlSifreIslemleri.Controls.Add(this.label5);
219:    this.pnlSifreIslemleri.Location = new System.Drawing.Point(411, 56);
220:    this.pnlSifreIslemleri.Name = "pnlSifreIslemleri";
221:    this.pnlSifreIslemleri.Size = new System.Drawing.Size(377, 181);
222:    this.pnlSifreIslemleri.TabIndex = 3;
223:    //
224:    // label7
225:    //
226:    this.label7.AutoSize = true;
227:    this.label7.Dock = System.Windows.Forms.DockStyle.Top;
228:    this.label7.Location = new System.Drawing.Point(0, 0);
229:    this.label7.Name = "label7";
230:    this.label7.Size = new System.Drawing.Size(68, 13);
```

```
231:         this.label7.TabIndex = 18;
232:         this.label7.Text = "Sifre Islemleri";
233:         this.label7.TextAlign = System.Drawing.ContentAlignment.TopCenter;
234:         //
235:         // btnSifreUygunMu
236:         //
237:         this.btnSifreUygunMu.Location = new System.Drawing.Point(173, 71);
238:         this.btnSifreUygunMu.Name = "btnSifreUygunMu";
239:         this.btnSifreUygunMu.Size = new System.Drawing.Size(145, 23);
240:         this.btnSifreUygunMu.TabIndex = 17;
241:         this.btnSifreUygunMu.Text = "Sifre Uygun Mu?";
242:         this.btnSifreUygunMu.UseVisualStyleBackColor = true;
243:         this.btnSifreUygunMu.Click += new System.EventHandler(this.btnSifreUygunMu_Click);
244:         //
245:         // btnRastgeleSifreUret
246:         //
247:         this.btnRastgeleSifreUret.Location = new System.Drawing.Point(21, 71);
248:         this.btnRastgeleSifreUret.Name = "btnRastgeleSifreUret";
249:         this.btnRastgeleSifreUret.Size = new System.Drawing.Size(145, 23);
250:         this.btnRastgeleSifreUret.TabIndex = 16;
251:         this.btnRastgeleSifreUret.Text = "Rastgele Sifre Üret";
252:         this.btnRastgeleSifreUret.UseVisualStyleBackColor = true;
253:         this.btnRastgeleSifreUret.Click += new
System.EventHandler(this.btnRastgeleSifreUret_Click);
254:         //
255:         // btnSha512Hash
256:         //
257:         this.btnSha512Hash.Location = new System.Drawing.Point(243, 42);
258:         this.btnSha512Hash.Name = "btnSha512Hash";
259:         this.btnSha512Hash.Size = new System.Drawing.Size(75, 23);
260:         this.btnSha512Hash.TabIndex = 15;
261:         this.btnSha512Hash.Text = "SHA 512";
262:         this.btnSha512Hash.UseVisualStyleBackColor = true;
263:         this.btnSha512Hash.Click += new System.EventHandler(this.btnSha512Hash_Click);
264:         //
265:         // btnSha256Hash
266:         //
267:         this.btnSha256Hash.Location = new System.Drawing.Point(131, 42);
268:         this.btnSha256Hash.Name = "btnSha256Hash";
269:         this.btnSha256Hash.Size = new System.Drawing.Size(75, 23);
270:         this.btnSha256Hash.TabIndex = 14;
271:         this.btnSha256Hash.Text = "SHA 256";
272:         this.btnSha256Hash.UseVisualStyleBackColor = true;
273:         this.btnSha256Hash.Click += new System.EventHandler(this.btnSha256Hash_Click);
274:         //
275:         // label6
276:         //
277:         this.label6.AutoSize = true;
278:         this.label6.Location = new System.Drawing.Point(21, 100);
279:         this.label6.Name = "label6";
280:         this.label6.Size = new System.Drawing.Size(38, 13);
281:         this.label6.TabIndex = 13;
282:         this.label6.Text = "Sonuç";
283:         //
284:         // lbSifreSonuc
285:         //
286:         this.lbSifreSonuc.FormattingEnabled = true;
287:         this.lbSifreSonuc.HorizontalScrollbar = true;
288:         this.lbSifreSonuc.Location = new System.Drawing.Point(65, 100);
289:         this.lbSifreSonuc.Name = "lbSifreSonuc";
290:         this.lbSifreSonuc.Size = new System.Drawing.Size(251, 69);
291:         this.lbSifreSonuc.TabIndex = 12;
292:         //
293:         // btnMd5Hash
294:         //
295:         this.btnMd5Hash.Location = new System.Drawing.Point(21, 42);
296:         this.btnMd5Hash.Name = "btnMd5Hash";
297:         this.btnMd5Hash.Size = new System.Drawing.Size(75, 23);
298:         this.btnMd5Hash.TabIndex = 11;
```

```

299:         this.btnMd5Hash.Text = "MD5";
300:         this.btnMd5Hash.UseVisualStyleBackColor = true;
301:         this.btnMd5Hash.Click += new System.EventHandler(this.btnMd5Hash_Click);
302:         //
303:         // txtSifrelenecekMetin
304:         //
305:         this.txtSifrelenecekMetin.Location = new System.Drawing.Point(118, 16);
306:         this.txtSifrelenecekMetin.Name = "txtSifrelenecekMetin";
307:         this.txtSifrelenecekMetin.Size = new System.Drawing.Size(199, 20);
308:         this.txtSifrelenecekMetin.TabIndex = 4;
309:         //
310:         // label5
311:         //
312:         this.label5.AutoSize = true;
313:         this.label5.Location = new System.Drawing.Point(17, 19);
314:         this.label5.Name = "label5";
315:         this.label5.Size = new System.Drawing.Size(95, 13);
316:         this.label5.TabIndex = 3;
317:         this.label5.Text = "Sifrelenecek Metin";
318:         //
319:         // pnlSayiIslemleri
320:         //
321:         this.pnlSayiIslemleri.BorderStyle = System.Windows.Forms.BorderStyle.FixedSingle;
322:         this.pnlSayiIslemleri.Controls.Add(this.label9);
323:         this.pnlSayiIslemleri.Controls.Add(this.btnOnayKoduOlustur);
324:         this.pnlSayiIslemleri.Controls.Add(this.btnMetneDonustur);
325:         this.pnlSayiIslemleri.Controls.Add(this.label10);
326:         this.pnlSayiIslemleri.Controls.Add(this.lbSayiSonuc);
327:         this.pnlSayiIslemleri.Controls.Add(this.btnSayisalMi);
328:         this.pnlSayiIslemleri.Controls.Add(this.txtSayi);
329:         this.pnlSayiIslemleri.Controls.Add(this.label11);
330:         this.pnlSayiIslemleri.Location = new System.Drawing.Point(15, 257);
331:         this.pnlSayiIslemleri.Name = "pnlSayiIslemleri";
332:         this.pnlSayiIslemleri.Size = new System.Drawing.Size(377, 181);
333:         this.pnlSayiIslemleri.TabIndex = 19;
334:         //
335:         // label9
336:         //
337:         this.label9.AutoSize = true;
338:         this.label9.Dock = System.Windows.Forms.DockStyle.Top;
339:         this.label9.Location = new System.Drawing.Point(0, 0);
340:         this.label9.Name = "label9";
341:         this.label9.Size = new System.Drawing.Size(67, 13);
342:         this.label9.TabIndex = 18;
343:         this.label9.Text = "Sayi Islemleri";
344:         this.label9.TextAlign = System.Drawing.ContentAlignment.TopCenter;
345:         //
346:         // btnOnayKoduOlustur
347:         //
348:         this.btnOnayKoduOlustur.Location = new System.Drawing.Point(243, 42);
349:         this.btnOnayKoduOlustur.Name = "btnOnayKoduOlustur";
350:         this.btnOnayKoduOlustur.Size = new System.Drawing.Size(75, 23);
351:         this.btnOnayKoduOlustur.TabIndex = 15;
352:         this.btnOnayKoduOlustur.Text = "Onay Kodu";
353:         this.btnOnayKoduOlustur.UseVisualStyleBackColor = true;
354:         this.btnOnayKoduOlustur.Click += new
System.EventHandler(this.btnOnayKoduOlustur_Click);
355:         //
356:         // btnMetneDonustur
357:         //
358:         this.btnMetneDonustur.Location = new System.Drawing.Point(102, 42);
359:         this.btnMetneDonustur.Name = "btnMetneDonustur";
360:         this.btnMetneDonustur.Size = new System.Drawing.Size(135, 23);
361:         this.btnMetneDonustur.TabIndex = 14;
362:         this.btnMetneDonustur.Text = "Para Metnine Dönüştür";
363:         this.btnMetneDonustur.UseVisualStyleBackColor = true;
364:         this.btnMetneDonustur.Click += new
System.EventHandler(this.btnMetneDonustur_Click);
365:         //

```

```
366: // label10
367: //
368: this.label10.AutoSize = true;
369: this.label10.Location = new System.Drawing.Point(18, 78);
370: this.label10.Name = "label10";
371: this.label10.Size = new System.Drawing.Size(38, 13);
372: this.label10.TabIndex = 13;
373: this.label10.Text = "Sonuç";
374: //
375: // lbSayiSonuc
376: //
377: this.lbSayiSonuc.FormattingEnabled = true;
378: this.lbSayiSonuc.HorizontalScrollbar = true;
379: this.lbSayiSonuc.Location = new System.Drawing.Point(62, 78);
380: this.lbSayiSonuc.Name = "lbSayiSonuc";
381: this.lbSayiSonuc.Size = new System.Drawing.Size(254, 82);
382: this.lbSayiSonuc.TabIndex = 12;
383: //
384: // btnSayisalMi
385: //
386: this.btnSayisalMi.Location = new System.Drawing.Point(21, 42);
387: this.btnSayisalMi.Name = "btnSayisalMi";
388: this.btnSayisalMi.Size = new System.Drawing.Size(75, 23);
389: this.btnSayisalMi.TabIndex = 11;
390: this.btnSayisalMi.Text = "Sayisal Mi?";
391: this.btnSayisalMi.UseVisualStyleBackColor = true;
392: this.btnSayisalMi.Click += new System.EventHandler(this.btnSayisalMi_Click);
393: //
394: // txtSayi
395: //
396: this.txtSayi.Location = new System.Drawing.Point(118, 16);
397: this.txtSayi.Name = "txtSayi";
398: this.txtSayi.Size = new System.Drawing.Size(199, 20);
399: this.txtSayi.TabIndex = 4;
400: //
401: // label11
402: //
403: this.label11.AutoSize = true;
404: this.label11.Location = new System.Drawing.Point(17, 19);
405: this.label11.Name = "label11";
406: this.label11.Size = new System.Drawing.Size(95, 13);
407: this.label11.TabIndex = 3;
408: this.label11.Text = "Sifrelenecek Metin";
409: //
410: // pnlKisiIslemleri
411: //
412: this.pnlKisiIslemleri.BorderStyle = System.Windows.Forms.BorderStyle.FixedSingle;
413: this.pnlKisiIslemleri.Controls.Add(this.label12);
414: this.pnlKisiIslemleri.Controls.Add(this.btnEPostaDogrula);
415: this.pnlKisiIslemleri.Controls.Add(this.btnIbanDogrula);
416: this.pnlKisiIslemleri.Controls.Add(this.label13);
417: this.pnlKisiIslemleri.Controls.Add(this.lbKisiSonuc);
418: this.pnlKisiIslemleri.Controls.Add(this.btnTcKimlikDogrula);
419: this.pnlKisiIslemleri.Controls.Add(this.txtKisiMetni);
420: this.pnlKisiIslemleri.Controls.Add(this.label14);
421: this.pnlKisiIslemleri.Location = new System.Drawing.Point(411, 257);
422: this.pnlKisiIslemleri.Name = "pnlKisiIslemleri";
423: this.pnlKisiIslemleri.Size = new System.Drawing.Size(377, 181);
424: this.pnlKisiIslemleri.TabIndex = 20;
425: //
426: // label12
427: //
428: this.label12.AutoSize = true;
429: this.label12.Dock = System.Windows.Forms.DockStyle.Top;
430: this.label12.Location = new System.Drawing.Point(0, 0);
431: this.label12.Name = "label12";
432: this.label12.Size = new System.Drawing.Size(63, 13);
433: this.label12.TabIndex = 18;
434: this.label12.Text = "Kisi Islemleri";
```



```
435:         this.label12.TextAlign = System.Drawing.ContentAlignment.TopCenter;
436:         //
437:         // btnEPostaDogrula
438:         //
439:         this.btnEPostaDogrula.Location = new System.Drawing.Point(223, 42);
440:         this.btnEPostaDogrula.Name = "btnEPostaDogrula";
441:         this.btnEPostaDogrula.Size = new System.Drawing.Size(95, 23);
442:         this.btnEPostaDogrula.TabIndex = 15;
443:         this.btnEPostaDogrula.Text = "E-Posta Dogrula";
444:         this.btnEPostaDogrula.UseVisualStyleBackColor = true;
445:         this.btnEPostaDogrula.Click += new
System.EventHandler(this.btnEPostaDogrula_Click);
446:         //
447:         // btnIbanDogrula
448:         //
449:         this.btnIbanDogrula.Location = new System.Drawing.Point(131, 42);
450:         this.btnIbanDogrula.Name = "btnIbanDogrula";
451:         this.btnIbanDogrula.Size = new System.Drawing.Size(86, 23);
452:         this.btnIbanDogrula.TabIndex = 14;
453:         this.btnIbanDogrula.Text = "IBAN Dogrula";
454:         this.btnIbanDogrula.UseVisualStyleBackColor = true;
455:         this.btnIbanDogrula.Click += new System.EventHandler(this.btnIbanDogrula_Click);
456:         //
457:         // label13
458:         //
459:         this.label13.AutoSize = true;
460:         this.label13.Location = new System.Drawing.Point(18, 78);
461:         this.label13.Name = "label13";
462:         this.label13.Size = new System.Drawing.Size(38, 13);
463:         this.label13.TabIndex = 13;
464:         this.label13.Text = "Sonuç";
465:         //
466:         // lbKisiSonuc
467:         //
468:         this.lbKisiSonuc.FormattingEnabled = true;
469:         this.lbKisiSonuc.HorizontalScrollbar = true;
470:         this.lbKisiSonuc.Location = new System.Drawing.Point(62, 78);
471:         this.lbKisiSonuc.Name = "lbKisiSonuc";
472:         this.lbKisiSonuc.Size = new System.Drawing.Size(254, 82);
473:         this.lbKisiSonuc.TabIndex = 12;
474:         //
475:         // btnTcKimlikDogrula
476:         //
477:         this.btnTcKimlikDogrula.Location = new System.Drawing.Point(21, 42);
478:         this.btnTcKimlikDogrula.Name = "btnTcKimlikDogrula";
479:         this.btnTcKimlikDogrula.Size = new System.Drawing.Size(104, 23);
480:         this.btnTcKimlikDogrula.TabIndex = 11;
481:         this.btnTcKimlikDogrula.Text = "TC Kimlik Dogrula";
482:         this.btnTcKimlikDogrula.UseVisualStyleBackColor = true;
483:         this.btnTcKimlikDogrula.Click += new
System.EventHandler(this.btnTcKimlikDogrula_Click);
484:         //
485:         // txtKisiMetni
486:         //
487:         this.txtKisiMetni.Location = new System.Drawing.Point(118, 16);
488:         this.txtKisiMetni.Name = "txtKisiMetni";
489:         this.txtKisiMetni.Size = new System.Drawing.Size(199, 20);
490:         this.txtKisiMetni.TabIndex = 4;
491:         //
492:         // label14
493:         //
494:         this.label14.AutoSize = true;
495:         this.label14.Location = new System.Drawing.Point(17, 19);
496:         this.label14.Name = "label14";
497:         this.label14.Size = new System.Drawing.Size(33, 13);
498:         this.label14.TabIndex = 3;
499:         this.label14.Text = "Metin";
500:         //
501:         // pnlMetinIslemleri
```

```

502:    //
503:    this.pnlMetinIslemleri.BorderStyle = System.Windows.Forms.BorderStyle.FixedSingle;
504:    this.pnlMetinIslemleri.Controls.Add(this.btnTelefonNoDuzenle);
505:    this.pnlMetinIslemleri.Controls.Add(this.btnMetinSifrele);
506:    this.pnlMetinIslemleri.Controls.Add(this.label15);
507:    this.pnlMetinIslemleri.Controls.Add(this.btnTürkçeKarakterDuzelt);
508:    this.pnlMetinIslemleri.Controls.Add(this.btnIlkHarfiBuyukYap);
509:    this.pnlMetinIslemleri.Controls.Add(this.label16);
510:    this.pnlMetinIslemleri.Controls.Add(this.lbMetinSonuc);
511:    this.pnlMetinIslemleri.Controls.Add(this.btnIlkHarfleriBuyukYap);
512:    this.pnlMetinIslemleri.Controls.Add(this.txtMetin);
513:    this.pnlMetinIslemleri.Controls.Add(this.label17);
514:    this.pnlMetinIslemleri.Location = new System.Drawing.Point(15, 458);
515:    this.pnlMetinIslemleri.Name = "pnlMetinIslemleri";
516:    this.pnlMetinIslemleri.Size = new System.Drawing.Size(773, 181);
517:    this.pnlMetinIslemleri.TabIndex = 21;
518:    //
519:    // btnTelefonNoDuzenle
520:    //
521:    this.btnTelefonNoDuzenle.Location = new System.Drawing.Point(503, 42);
522:    this.btnTelefonNoDuzenle.Name = "btnTelefonNoDuzenle";
523:    this.btnTelefonNoDuzenle.Size = new System.Drawing.Size(134, 23);
524:    this.btnTelefonNoDuzenle.TabIndex = 20;
525:    this.btnTelefonNoDuzenle.Text = "Telefon No Düzenle";
526:    this.btnTelefonNoDuzenle.UseVisualStyleBackColor = true;
527:    this.btnTelefonNoDuzenle.Click += new
System.EventHandler(this.btnTelefonNoDuzenle_Click);
528:    //
529:    // btnMetinSifrele
530:    //
531:    this.btnMetinSifrele.Location = new System.Drawing.Point(363, 42);
532:    this.btnMetinSifrele.Name = "btnMetinSifrele";
533:    this.btnMetinSifrele.Size = new System.Drawing.Size(134, 23);
534:    this.btnMetinSifrele.TabIndex = 19;
535:    this.btnMetinSifrele.Text = "Metin Sifrele";
536:    this.btnMetinSifrele.UseVisualStyleBackColor = true;
537:    this.btnMetinSifrele.Click += new System.EventHandler(this.btnMetinSifrele_Click);
538:    //
539:    // label15
540:    //
541:    this.label15.AutoSize = true;
542:    this.label15.Dock = System.Windows.Forms.DockStyle.Top;
543:    this.label15.Location = new System.Drawing.Point(0, 0);
544:    this.label15.Name = "label15";
545:    this.label15.Size = new System.Drawing.Size(73, 13);
546:    this.label15.TabIndex = 18;
547:    this.label15.Text = "Metin Islemleri";
548:    this.label15.TextAlign = System.Drawing.ContentAlignment.TopCenter;
549:    //
550:    // btnTürkçeKarakterDuzelt
551:    //
552:    this.btnTürkçeKarakterDuzelt.Location = new System.Drawing.Point(223, 42);
553:    this.btnTürkçeKarakterDuzelt.Name = "btnTürkçeKarakterDuzelt";
554:    this.btnTürkçeKarakterDuzelt.Size = new System.Drawing.Size(134, 23);
555:    this.btnTürkçeKarakterDuzelt.TabIndex = 15;
556:    this.btnTürkçeKarakterDuzelt.Text = "Türkçe Karakter Düzelt";
557:    this.btnTürkçeKarakterDuzelt.UseVisualStyleBackColor = true;
558:    this.btnTürkçeKarakterDuzelt.Click += new
System.EventHandler(this.btnTürkçeKarakterDuzelt_Click);
559:    //
560:    // btnIlkHarfiBuyukYap
561:    //
562:    this.btnIlkHarfiBuyukYap.Location = new System.Drawing.Point(131, 42);
563:    this.btnIlkHarfiBuyukYap.Name = "btnIlkHarfiBuyukYap";
564:    this.btnIlkHarfiBuyukYap.Size = new System.Drawing.Size(86, 23);
565:    this.btnIlkHarfiBuyukYap.TabIndex = 14;
566:    this.btnIlkHarfiBuyukYap.Text = "Ilk Harf Büyük";
567:    this.btnIlkHarfiBuyukYap.UseVisualStyleBackColor = true;
568:    this.btnIlkHarfiBuyukYap.Click += new

```

```

System.EventHandler(this.btnIlkHarfiBuyukYap_Click);
569:    //
570:    // label16
571:    //
572:    this.label16.AutoSize = true;
573:    this.label16.Location = new System.Drawing.Point(18, 78);
574:    this.label16.Name = "label16";
575:    this.label16.Size = new System.Drawing.Size(38, 13);
576:    this.label16.TabIndex = 13;
577:    this.label16.Text = "Sonuç";
578:    //
579:    // lbMetinSonuc
580:    //
581:    this.lbMetinSonuc.FormattingEnabled = true;
582:    this.lbMetinSonuc.HorizontalScrollbar = true;
583:    this.lbMetinSonuc.Location = new System.Drawing.Point(62, 78);
584:    this.lbMetinSonuc.Name = "lbMetinSonuc";
585:    this.lbMetinSonuc.Size = new System.Drawing.Size(652, 82);
586:    this.lbMetinSonuc.TabIndex = 12;
587:    //
588:    // btnIlkHarfleriBuyukYap
589:    //
590:    this.btnIlkHarfleriBuyukYap.Location = new System.Drawing.Point(21, 42);
591:    this.btnIlkHarfleriBuyukYap.Name = "btnIlkHarfleriBuyukYap";
592:    this.btnIlkHarfleriBuyukYap.Size = new System.Drawing.Size(104, 23);
593:    this.btnIlkHarfleriBuyukYap.TabIndex = 11;
594:    this.btnIlkHarfleriBuyukYap.Text = "Ilk Harfler Büyük";
595:    this.btnIlkHarfleriBuyukYap.UseVisualStyleBackColor = true;
596:    this.btnIlkHarfleriBuyukYap.Click += new
System.EventHandler(this.btnIlkHarfleriBuyukYap_Click);
597:    //
598:    // txtMetin
599:    //
600:    this.txtMetin.Location = new System.Drawing.Point(119, 12);
601:    this.txtMetin.Name = "txtMetin";
602:    this.txtMetin.Size = new System.Drawing.Size(595, 20);
603:    this.txtMetin.TabIndex = 4;
604:    //
605:    // label17
606:    //
607:    this.label17.AutoSize = true;
608:    this.label17.Location = new System.Drawing.Point(17, 19);
609:    this.label17.Name = "label17";
610:    this.label17.Size = new System.Drawing.Size(33, 13);
611:    this.label17.TabIndex = 3;
612:    this.label17.Text = "Metin";
613:    //
614:    // FrmAnaSayfa
615:    //
616:    this.AutoScaleDimensions = new System.Drawing.SizeF(6F, 13F);
617:    this.AutoScaleMode = System.Windows.Forms.AutoScaleModeMode.Font;
618:    this.AutoSizeMode = System.Windows.Forms.AutoSizeMode.GrowAndShrink;
619:    this.ClientSize = new System.Drawing.Size(800, 686);
620:    this.Controls.Add(this.pnlMetinIslemleri);
621:    this.Controls.Add(this.pnlKisiIslemleri);
622:    this.Controls.Add(this.pnlSayiIslemleri);
623:    this.Controls.Add(this.pnlSifreIslemleri);
624:    this.Controls.Add(this.pnlTarihIslemleri);
625:    this.Controls.Add(this.label1);
626:    this.Controls.Add(this.cmbIslemTuru);
627:    this.Name = "FrmAnaSayfa";
628:    this.StartPosition = System.Windows.Forms.FormStartPosition.CenterScreen;
629:    this.Text = "Islemler Ekrani";
630:    this.Load += new System.EventHandler(this.FrmAnaSayfa_Load);
631:    this.pnlTarihIslemleri.ResumeLayout(false);
632:    this.pnlTarihIslemleri.PerformLayout();
633:    this.pnlSifreIslemleri.ResumeLayout(false);
634:    this.pnlSifreIslemleri.PerformLayout();
635:    this.pnlSayiIslemleri.ResumeLayout(false);

```



```

636:         this.pnlSayiIslemleri.PerformLayout();
637:         this.pnlKisiIslemleri.ResumeLayout(false);
638:         this.pnlKisiIslemleri.PerformLayout();
639:         this.pnlMetinIslemleri.ResumeLayout(false);
640:         this.pnlMetinIslemleri.PerformLayout();
641:         this.ResumeLayout(false);
642:         this.PerformLayout();
643:     }

```

## 1.1.3.2 Program Class

### Class Hierarchy

CSharpHelperMethods.Program

### C#

```
internal static class Program;
```

### File

Program.cs (see page 94)

### Description

This is class CSharpHelperMethods.Program.

### Program Methods

	Name	Description
	Main (see page 62)	The main entry point for the application.

## 1.1.3.2.1 Program Methods

### 1.1.3.2.1.1 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 63)	This is file AssemblyInfo.cs.

CSharpHelperMethods.csproj (🔗 see page 64)	This is file CSharpHelperMethods.csproj.
CSharpHelperMethods.sln (🔗 see page 64)	This is file CSharpHelperMethods.sln.
DataTableIslemleri.cs (🔗 see page 64)	This is file DataTableIslemleri.cs.
Dosyalslemleri.cs (🔗 see page 66)	This is file Dosyalslemleri.cs.
FrmAnaSayfa.cs (🔗 see page 68)	This is file FrmAnaSayfa.cs.
FrmAnaSayfa.Designer.cs (🔗 see page 78)	This is file FrmAnaSayfa.Designer.cs.
Kisilslemleri.cs (🔗 see page 89)	This is file Kisilslemleri.cs.
MetinIslemleri.cs (🔗 see page 91)	This is file MetinIslemleri.cs.
Program.cs (🔗 see page 94)	This is file Program.cs.
Resources.Designer.cs (🔗 see page 94)	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.
SayIslemleri.cs (🔗 see page 96)	This is file SayIslemleri.cs.
Settings.Designer.cs (🔗 see page 98)	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.
Sifrelemelslemleri.cs (🔗 see page 99)	This is file Sifrelemelslemleri.cs.
Siralamalslemleri.cs (🔗 see page 101)	This is file Siralamalslemleri.cs.
Tarihslemleri.cs (🔗 see page 102)	This is file Tarihslemleri.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")]
8: [assembly: AssemblyDescription("")]
9: [assembly: AssemblyConfiguration("")]
10: [assembly: AssemblyCompany("")]
11: [assembly: AssemblyProduct("CSharpHelperMethods")]
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.csproj

This is file CSharpHelperMethods.csproj.

## 1.2.3 CSharpHelperMethods.sln

This is file CSharpHelperMethods.sln.

## 1.2.4 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.YardimciSiniflar
7: {
8:     /// <summary>
9:     /// Raporlama için لازم olan DataTable vb. verilerin düzenlenmesini saglayan
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
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 |
27:                BindingFlags.CreateInstance | BindingFlags.Default | BindingFlags.DeclaredOnly |
28:                BindingFlags.ExactBinding | BindingFlags.FlattenHierarchy | BindingFlags.IgnoreReturn |
29:                BindingFlags.Instance | BindingFlags.InvokeMethod | BindingFlags.NonPublic |
30:                BindingFlags.OptionalParamBinding | BindingFlags.SuppressChangeType);
31:
32:                var ozellikler2 = typeof(T).GetProperties(BindingFlags.Public |
33:                BindingFlags.CreateInstance | BindingFlags.Default | BindingFlags.DeclaredOnly |
34:                BindingFlags.ExactBinding | BindingFlags.FlattenHierarchy | BindingFlags.IgnoreReturn |
35:                BindingFlags.Instance | BindingFlags.InvokeMethod | BindingFlags.NonPublic |
36:                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:             sonuc.Rows.Add(degerler);
82:
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.

## 1.2.5 Dosyalslemleri.cs

This is file Dosyalslemleri.cs.

### Body Source

```

1: ?using System;
2: using System.IO;
3:
4: namespace CSharpHelperMethods.YardimciSiniflar
5: {
6:     /// <summary>
7:     /// Dosya islemleri ile ilgili yardimci sinif
8:     /// </summary>
9:     public static class DosyaIslemleri
10:    {
11:        /// <summary>
12:        /// Dosya uzantisinden 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 GetIcerikTipiDosyaUzantisiIle(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
79:     /// Dosyanin uzantisi istedigimiz formatta degilse yanlis olarak geri
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 tarafından o anda
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,
FileStream.None);
101:             }
102:             catch (IOException)
103:             {
104:                 // eger dosya baska bir uygulama tarafindan kullaniliyorsa
105:                 // burada true dönüyor ki
106:                 // kullanıcı bu metottan gelen bilgiye göre davranisin
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 gönderdiğiniz Dizin yolunda verdiğiniz isimde bir klasör yoksa
120:         /// Otomatik olarak olusturmayı 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.

## 1.2.6 FrmAnaSayfa.cs

This is file FrmAnaSayfa.cs.

## Body Source

```

1: ?using System.Drawing;
2: using CSharpHelperMethods.YardimciSiniflar;
3: using System.Text;
4: using System.Windows.Forms;
5:
6: namespace CSharpHelperMethods
7: {
8:     public partial class FrmAnaSayfa : Form
9:     {
10:         public FrmAnaSayfa()
11:         {
12:             InitializeComponent();
13:         }
14:

```

```

15:         #region Formun Yüklmesi, Panel Seçimi ve ListBox Düzenlemeleri
16:         /// <summary>
17:         /// Form İlk Yüklendiği zaman ekrandaki bütün panellerin erişilemez hale
getirilmesi
18:         /// sağlanıyor Bu metotta, çünkü hangi panelin aktifleceğinin
19:         /// Combobox üzerinden seçilmesi lazım
20:         /// </summary>
21:         /// <param name="sender">The sender info (For example Main Form)</param>
22:         /// <param name="e">Event Arguments</param>
23:         private void FrmAnaSayfa_Load(object sender, System.EventArgs e)
24:         {
25:             cmbIslemTuru.DropDownStyle = ComboBoxStyle.DropDownList;
26:             #region Sayfa İlk Yüklendiğinde Tüm Panellerin Erisime Kapanması
27:             pnlTarihIslemleri.Enabled = false;
28:             pnlSifreIslemleri.Enabled = false;
29:             pnlSayiIslemleri.Enabled = false;
30:             pnlKisiIslemleri.Enabled = false;
31:             pnlMetinIslemleri.Enabled = false;
32:             #endregion
33:             #region Bütün Listboxların MultiLine Yapılması
34:             lbMetinSonuc.DrawMode = DrawMode.OwnerDrawVariable;
35:             lbMetinSonuc.MeasureItem += listBox_MeasureItem;
36:             lbMetinSonuc.DrawItem += listBox_DrawItem;
37:             lbSayiSonuc.DrawMode = DrawMode.OwnerDrawVariable;
38:             lbSayiSonuc.MeasureItem += listBox_MeasureItem;
39:             lbSayiSonuc.DrawItem += listBox_DrawItem;
40:             lbKisiSonuc.DrawMode = DrawMode.OwnerDrawVariable;
41:             lbKisiSonuc.MeasureItem += listBox_MeasureItem;
42:             lbKisiSonuc.DrawItem += listBox_DrawItem;
43:             lbSifreSonuc.DrawMode = DrawMode.OwnerDrawVariable;
44:             lbSifreSonuc.MeasureItem += listBox_MeasureItem;
45:             lbSifreSonuc.DrawItem += listBox_DrawItem;
46:             lbSonuc.DrawMode = DrawMode.OwnerDrawVariable;
47:             lbSonuc.MeasureItem += listBox_MeasureItem;
48:             lbSonuc.DrawItem += listBox_DrawItem;
49:             #endregion
50:         }
51:
52:         /// <summary>
53:         /// Islem Türü Combobox'ında seçilen degere göre
54:         /// Ekrandaki panellerden ilgili olan haricinde hepsinin erişilemez hale
getirilmesi
55:         /// Eger bir sey seçilmemisse, tamaminin erişilemez hale getirilmesi için
56:         /// Gerekli işlemlerin gerçekleştirildiği metottur.
57:         /// </summary>
58:         /// <param name="sender">The sender info (For example Main Form)</param>
59:         /// <param name="e">Event Arguments</param>
60:         private void cmbIslemTuru_SelectedIndexChanged(object sender, System.EventArgs
e)
61:         {
62:             #region Combobox'tan Seçilen Degere Göre İlgili Paneli Aktif Hale Getirme
63:             switch (cmbIslemTuru.SelectedIndex)
64:             {
65:                 // Metin İşlemleri
66:                 case 0:
67:                 {
68:                     pnlMetinIslemleri.Enabled = true;
69:                     pnlTarihIslemleri.Enabled = false;
70:                     pnlSifreIslemleri.Enabled = false;
71:                     pnlSayiIslemleri.Enabled = false;
72:                     pnlKisiIslemleri.Enabled = false;
73:                     break;
74:                 }
75:                 // Tarih İşlemleri
76:                 case 1:
77:                 {
78:                     pnlMetinIslemleri.Enabled = false;
79:                     pnlTarihIslemleri.Enabled = true;
80:                     pnlSifreIslemleri.Enabled = false;

```



```

81:                 pnlSayiIslemleri.Enabled = false;
82:                 pnlKisiIslemleri.Enabled = false;
83:                 break;
84:             }
85:             // Sayi Islemleri
86:             case 2:
87:             {
88:                 pnlMetinIslemleri.Enabled = false;
89:                 pnlTarihIslemleri.Enabled = false;
90:                 pnlSifreIslemleri.Enabled = false;
91:                 pnlSayiIslemleri.Enabled = true;
92:                 pnlKisiIslemleri.Enabled = false;
93:                 break;
94:             }
95:             // Sifre Islemleri
96:             case 3:
97:             {
98:                 pnlMetinIslemleri.Enabled = false;
99:                 pnlTarihIslemleri.Enabled = false;
100:                 pnlSifreIslemleri.Enabled = true;
101:                 pnlSayiIslemleri.Enabled = false;
102:                 pnlKisiIslemleri.Enabled = false;
103:                 break;
104:             }
105:             // Kisi Islemleri
106:             case 4:
107:             {
108:                 pnlMetinIslemleri.Enabled = false;
109:                 pnlTarihIslemleri.Enabled = false;
110:                 pnlSifreIslemleri.Enabled = false;
111:                 pnlSayiIslemleri.Enabled = false;
112:                 pnlKisiIslemleri.Enabled = true;
113:                 break;
114:             }
115:             // Bunlar haricinde herhangi bir sey
116:             default:
117:             {
118:                 pnlMetinIslemleri.Enabled = false;
119:                 pnlTarihIslemleri.Enabled = false;
120:                 pnlSifreIslemleri.Enabled = false;
121:                 pnlSayiIslemleri.Enabled = false;
122:                 pnlKisiIslemleri.Enabled = false;
123:                 break;
124:             }
125:         }
126:     #endregion
127: }
128:
129:     /// <summary>
130:     /// This method is used to measure the length of the item in listbox and make
it multi line
131:     /// without using scrollbar and let the user see more efficiently
132:     /// </summary>
133:     /// <param name="sender">The sender info (For example Main Form)</param>
134:     /// <param name="e">Event Arguments</param>
135:     private static void listBox_MeasureItem(object sender, MeasureItemEventArgs e)
136:     {
137:         if (!(sender is ListBox listBox)) return;
138:         e.ItemHeight = (int) e.Graphics
139:             .MeasureString(listBox.Items[e.Index].ToString(), listBox.Font,
listBox.Width).Height;
140:     }
141:
142:     /// <summary>
143:     /// This method is used to measure the length of the item in listbox and make
it multi line
144:     /// without using scrollbar and let the user see more efficiently
145:     /// </summary>
146:     /// <param name="sender">The sender info (For example Main Form)</param>

```

```

147:         /// <param name="e">Event Arguments</param>
148:         private static void listBox_DrawItem(object sender, DrawItemEventArgs e)
149:         {
150:             if (!(sender is ListBox listBox)) return;
151:             e.DrawBackground();
152:             e.DrawFocusRectangle();
153:             e.Graphics.DrawString(listBox.Items[e.Index].ToString(), e.Font, new
SolidBrush(e.ForeColor), e.Bounds);
154:         }
155:
156:         #endregion
157:
158:         #region Tarih Islemleri
159:
160:         /// <summary>
161:         /// Paneldeki Baslangiç Tarihi seçili ve tarih girilirse,
162:         /// o tarihten bugüne kaç yas oldugunun hesaplandigi metottur.
163:         /// Ilgili butona tiklandigi zaman çalışır
164:         /// </summary>
165:         /// <param name="sender">The sender info (For example Main Form)</param>
166:         /// <param name="e">Event Arguments</param>
167:         private void btnYasHesapla_Click(object sender, System.EventArgs e)
168:         {
169:             if (!dtpBaslangicTarihi.Checked)
170:                 MessageBox.Show("Baslangiç Tarihi girilmedigi için bu işlem
gerçekleştiremez");
171:             else
172:             {
173:                 lbSonuc.Items.Clear();
174:                 var sb = new StringBuilder();
175:                 sb.Append("Dogum tarihi seçilen kisi ");
176:                 sb.Append(TarihIslemleri.YasHesapla(dtpBaslangicTarihi.Value));
177:                 sb.Append(" yasinin içindedir");
178:                 lbSonuc.Items.Add(sb.ToString());
179:             }
180:         }
181:
182:         /// <summary>
183:         /// Paneldeki Baslangiç Tarihi seçili ve tarih girilirse,
184:         /// o tarihten bugüne kaç yas oldugunun yıl, ay ve gün
185:         /// olarak metinsel bir şekilde hesaplandigi metottur.
186:         /// Ilgili butona tiklandigi zaman çalışır
187:         /// </summary>
188:         /// <param name="sender">The sender info (For example Main Form)</param>
189:         /// <param name="e">Event Arguments</param>
190:         private void btnYasHesaplaMetinsel_Click(object sender, System.EventArgs e)
191:         {
192:             if (!dtpBaslangicTarihi.Checked)
193:                 MessageBox.Show("Baslangiç Tarihi girilmedigi için bu işlem
gerçekleştiremez");
194:             else
195:             {
196:                 lbSonuc.Items.Clear();
197:                 var sb = new StringBuilder();
198:                 sb.Append(TarihIslemleri.YasHesaplaMetinsel(dtpBaslangicTarihi.Value));
199:                 lbSonuc.Items.Add(sb.ToString());
200:             }
201:         }
202:
203:         /// <summary>
204:         /// Paneldeki Baslangiç Tarihi ve Bitis Tarihi alanlari seçili ve tarih
girilirse,
205:         /// Söz konusu tarihler arasında kaç yıl, kaç ay ve gün oldugunun bilgisinin
hazirlandigi metottur.
206:         /// Ilgili butona tiklandigi zaman çalışır
207:         /// </summary>
208:         /// <param name="sender">The sender info (For example Main Form)</param>
209:         /// <param name="e">Event Arguments</param>
210:         private void btnTarihAraligi_Click(object sender, System.EventArgs e)

```

```

211:         {
212:             if (!dtpBaslangicTarihi.Checked || !dtpBitisTarihi.Checked)
213:                 MessageBox.Show("Baslangiç Tarihi veya Bitis Tarihi girilmediği için
bu işlem gerçekleştirilemez");
214:             else if (dtpBaslangicTarihi.Value.Date > dtpBitisTarihi.Value.Date)
215:                 MessageBox.Show("Baslangiç Tarihi Bitis Tarihinden büyük olamaz");
216:             else
217:             {
218:                 lbSonuc.Items.Clear();
219:                 var sb = new StringBuilder();
220:                 sb.Append("İki tarih arası: ");
221:                 sb.Append(TarihIslemleri.TarihAraligiHesapla(dtpBaslangicTarihi.Value.Date,
dtpBitisTarihi.Value.Date));
222:                 lbSonuc.Items.Add(sb.ToString());
223:             }
224:         }
225:
226:         /// <summary>
227:         /// Paneldeki Baslangiç Tarihi ve Bitis Tarihi alanları seçili ve tarih
girilmisse,
228:         /// Söz konusu tarihler arasında kaç ay olduğunun bilgisinin hazırlandığı
metottur.
229:         /// İlgili butona tıklandığı zaman çalışır
230:         /// </summary>
231:         /// <param name="sender">The sender info (For example Main Form)</param>
232:         /// <param name="e">Event Arguments</param>
233:         private void btnAyFarki_Click(object sender, System.EventArgs e)
234:         {
235:             if (!dtpBaslangicTarihi.Checked || !dtpBitisTarihi.Checked)
236:                 MessageBox.Show("Baslangiç Tarihi veya Bitis Tarihi girilmediği için
bu işlem gerçekleştirilemez");
237:             else if (dtpBaslangicTarihi.Value.Date > dtpBitisTarihi.Value.Date)
238:                 MessageBox.Show("Baslangiç Tarihi Bitis Tarihinden büyük olamaz");
239:             else
240:             {
241:                 lbSonuc.Items.Clear();
242:                 var sb = new StringBuilder();
243:                 sb.Append("Ay Farki: ");
244:                 sb.Append(TarihIslemleri.AyFarki(dtpBaslangicTarihi.Value.Date,
dtpBitisTarihi.Value.Date));
245:                 lbSonuc.Items.Add(sb.ToString());
246:             }
247:         }
248:         #endregion
249:
250:         #region Sifre Islemleri
251:         /// <summary>
252:         /// Paneldeki metin kutusu dolu ise
253:         /// Girilen metnin MD5 algoritması ile şifrelenmiş halini
254:         /// Listbox üzerine yazdıran metottur.
255:         /// </summary>
256:         /// <param name="sender">The sender info (For example Main Form)</param>
257:         /// <param name="e">Event Arguments</param>
258:         private void btnMd5Hash_Click(object sender, System.EventArgs e)
259:         {
260:             if (string.IsNullOrEmpty(txtSifrelenecekMetin.Text.Trim()))
261:                 MessageBox.Show("Sifrelenecek metin girilmediği için bu işlem
gerçekleştirilemez");
262:             else
263:             {
264:                 lbSifreSonuc.Items.Clear();
265:                 var sb = new StringBuilder();
266:                 sb.Append("Girilen Metnin MD5 Karsiligi: ");
267:                 sb.Append(SifrelemeIslemleri.HesaplaMD5(txtSifrelenecekMetin.Text.Trim()));
268:                 lbSifreSonuc.Items.Add(sb.ToString());
269:             }
270:         }

```

```

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

```

```

335:         private void btnSifreUygunMu_Click(object sender, System.EventArgs e)
336:         {
337:             if (string.IsNullOrEmpty(txtSifrelenecekMetin.Text.Trim()))
338:                 MessageBox.Show("Sifre metni girilmedigi için bu işlem
gerçekleştiremez");
339:             else
340:             {
341:                 lbSifreSonuc.Items.Clear();
342:                 var sb = new StringBuilder();
343:                 sb.Append("Girilen Sifrenin Uygunluk Durumu: ");
344:                 sb.Append(SifrelemeIslemleri.SifreUygunMu(txtSifrelenecekMetin.Text.Trim()));
345:                 lbSifreSonuc.Items.Add(sb.ToString());
346:             }
347:         }
348:     #endregion
349:
350:     #region Sayi Islemleri
351:     /// <summary>
352:     /// Paneldeki metin kutusuna girilen metnin sayısal bir değer olup olmadığını
353:     /// Kontrol eden ve sonucu gösteren metottur.
354:     /// </summary>
355:     /// <param name="sender">The sender info (For example Main Form)</param>
356:     /// <param name="e">Event Arguments</param>
357:     private void btnSayısalMi_Click(object sender, System.EventArgs e)
358:     {
359:         if (string.IsNullOrEmpty(txtSayi.Text.Trim()))
360:             MessageBox.Show("Metin girilmedigi için bu işlem gerçekleştirilemez");
361:         else
362:         {
363:             lbSayiSonuc.Items.Clear();
364:             var sb = new StringBuilder();
365:             sb.Append("Girilen Metin Sayı Mi?: ");
366:             sb.Append(SayıIslemleri.SayısalMi(txtSayi.Text.Trim()) ? "Evet" :
"Hayır");
367:             lbSayiSonuc.Items.Add(sb.ToString());
368:         }
369:     }
370:
371:     /// <summary>
372:     /// Paneldeki metin kutusuna girilen para birimi cinsinden rakamın
373:     /// Yazıya çevrilmesini sağlayan ve sonucu listbox üzerinde gösteren bir
metottur.
374:     /// </summary>
375:     /// <param name="sender">The sender info (For example Main Form)</param>
376:     /// <param name="e">Event Arguments</param>
377:     private void btnMetneDonustur_Click(object sender, System.EventArgs e)
378:     {
379:         if (string.IsNullOrEmpty(txtSayi.Text.Trim()))
380:             MessageBox.Show("Metin girilmedigi için bu işlem gerçekleştirilemez");
381:         else if (!SayıIslemleri.SayısalMi(txtSayi.Text.Trim().Replace(".", ""),
".").Replace(".", ""))
382:             MessageBox.Show("Girilen metin sayısal bir değer olmadığı için bu
işlem gerçekleştirilemez");
383:         else
384:         {
385:             var metinSadeceSayiKismi = txtSayi.Text.Trim();
386:             var sayi = SayıIslemleri.SayıHalineCevir(metinSadeceSayiKismi);
387:             lbSayiSonuc.Items.Clear();
388:             var sb = new StringBuilder();
389:             sb.Append("Girilen Sayının Metin Hali (Parasal): ");
390:             sb.Append(SayıIslemleri.SayıYazıyaCevir(sayi));
391:             lbSayiSonuc.Items.Add(sb.ToString());
392:         }
393:     }
394:
395:     /// <summary>
396:     /// Rastgele olacak şekilde 6 haneli kod oluşturmaya sağlayan bir metottur.
397:     /// </summary>

```

```

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

```

```

460:         /// </summary>
461:         /// <param name="sender">The sender info (For example Main Form)</param>
462:         /// <param name="e">Event Arguments</param>
463:         private void btnEPostaDogrula_Click(object sender, System.EventArgs e)
464:         {
465:             if (string.IsNullOrEmpty(txtKisiMetni.Text.Trim()))
466:                 MessageBox.Show("Metin girilmedigi için bu islem gerçekleştirilemez");
467:             else
468:             {
469:                 lbKisiSonuc.Items.Clear();
470:                 var sb = new StringBuilder();
471:                 sb.Append("E-Posta Dogru Mu?: ");
472:                 sb.Append(KisiIslemleri.DogrulaEPosta(txtKisiMetni.Text.Trim()) ?
"Ev" : "Hayir");
473:                 lbKisiSonuc.Items.Add(sb.ToString());
474:             }
475:         }
476:     #endregion
477:
478:     #region Metin Islemleri
479:     /// <summary>
480:     /// Paneldeki metin kutusuna girilen metnin
481:     /// Kelimelerinin ilk harflerini büyük yapmaya yarayan bir metottur.
482:     /// </summary>
483:     /// <param name="sender">The sender info (For example Main Form)</param>
484:     /// <param name="e">Event Arguments</param>
485:     private void btnIlkHarfleriBuyukYap_Click(object sender, System.EventArgs e)
486:     {
487:         if (string.IsNullOrEmpty(txtMetin.Text.Trim()))
488:             MessageBox.Show("Metin girilmedigi için bu islem gerçekleştirilemez");
489:         else
490:         {
491:             lbMetinSonuc.Items.Clear();
492:             var sb = new StringBuilder();
493:             sb.Append("Metnin Düzenlenmis Hali: ");
494:             sb.Append(txtMetin.Text.Trim().MetinIlkKarakterleriBuyukYap());
495:             lbMetinSonuc.Items.Add(sb.ToString());
496:         }
497:     }
498:
499:     /// <summary>
500:     /// Paneldeki metin kutusuna girilen metnin
501:     /// Sadece ilk harfini büyük yapmaya yarayan bir metottur.
502:     /// </summary>
503:     /// <param name="sender">The sender info (For example Main Form)</param>
504:     /// <param name="e">Event Arguments</param>
505:     private void btnIlkHarfiBuyukYap_Click(object sender, System.EventArgs e)
506:     {
507:         if (string.IsNullOrEmpty(txtMetin.Text.Trim()))
508:             MessageBox.Show("Metin girilmedigi için bu islem gerçekleştirilemez");
509:         else
510:         {
511:             lbMetinSonuc.Items.Clear();
512:             var sb = new StringBuilder();
513:             sb.Append("Metnin Düzenlenmis Hali: ");
514:             sb.Append(MetinIslemleri.MetinIlkKarakterBuyukDigerleriKucukYap(txtMetin.Text.Trim()));
515:             lbMetinSonuc.Items.Add(sb.ToString());
516:         }
517:     }
518:
519:     /// <summary>
520:     /// Paneldeki metin kutusuna girilen metnin
521:     /// İçerisindeki Türkçe karakterleri İngilizce hallerine çeviren bir metottur.
522:     /// </summary>
523:     /// <param name="sender">The sender info (For example Main Form)</param>
524:     /// <param name="e">Event Arguments</param>
525:     private void btnTurkceKarakterDuzelt_Click(object sender, System.EventArgs e)
526:     {

```



```

527:         if (string.IsNullOrEmpty(txtMetin.Text.Trim()))
528:             MessageBox.Show("Metin girilmediği için bu işlem gerçekleştirilemez");
529:         else
530:         {
531:             lbMetinSonuc.Items.Clear();
532:             var sb = new StringBuilder();
533:             sb.Append("Metnin Düzenlenmiş Hali: ");
534:
sb.Append(MetinIslemleri.TurkceKarakterleriDuzelt(txtMetin.Text.Trim()));
535:             lbMetinSonuc.Items.Add(sb.ToString());
536:         }
537:     }
538:
539:     /// <summary>
540:     /// KVKK durumlarından dolayı TC Kimlik No ve Vergi No gibi alanların
541:     /// Matbu çıktılarda gizlenmesi gibi bir durum söz konusu olduğundan
542:     /// Girilen bu 10 veya 11 karakterlik metinleri şifrelemeye yarayan bir
metottur
543:     /// </summary>
544:     /// <param name="sender">The sender info (For example Main Form)</param>
545:     /// <param name="e">Event Arguments</param>
546:     private void btnMetinSifrele_Click(object sender, System.EventArgs e)
547:     {
548:         if (string.IsNullOrEmpty(txtMetin.Text.Trim()))
549:             MessageBox.Show("Metin girilmediği için bu işlem gerçekleştirilemez");
550:         else if (!SayiIslemleri.SayisalMi(txtMetin.Text.Trim()))
551:             MessageBox.Show("Girilen metin sayısal bir değer olmadığı için bu
işlem gerçekleştirilemez");
552:         else if (txtMetin.Text.Trim().Length != 11 && txtMetin.Text.Trim().Length
!= 10)
553:             MessageBox.Show("Girilen metin 10 veya 11 karakter olmadığı için bu
işlem gerçekleştirilemez");
554:         else
555:         {
556:             lbMetinSonuc.Items.Clear();
557:             var sb = new StringBuilder();
558:             sb.Append("TC Kimlik No veya Vergi No'nun Şifrelenmiş Hali: ");
559:             sb.Append(MetinIslemleri.MetinSifrele(txtMetin.Text.Trim()));
560:             lbMetinSonuc.Items.Add(sb.ToString());
561:         }
562:     }
563:
564:     /// <summary>
565:     /// Ekrandan çeşitli karakterler içeren telefon numaraları alındığı zaman
566:     /// Bu telefon numarasını olması gerektiği hale çevirmeye yarayan bir metottur.
567:     /// </summary>
568:     /// <param name="sender">The sender info (For example Main Form)</param>
569:     /// <param name="e">Event Arguments</param>
570:     private void btnTelefonNoDuzenle_Click(object sender, System.EventArgs e)
571:     {
572:         if (string.IsNullOrEmpty(txtMetin.Text.Trim()))
573:             MessageBox.Show("Metin girilmediği için bu işlem gerçekleştirilemez");
574:         else
575:         {
576:             lbMetinSonuc.Items.Clear();
577:             var sb = new StringBuilder();
578:             sb.Append("Telefon Numarasının Düzenlenmiş Hali: ");
579:             sb.Append(MetinIslemleri.TelefonNoDuzenle(txtMetin.Text.Trim()));
580:             lbMetinSonuc.Items.Add(sb.ToString());
581:         }
582:     }
583:     #endregion
584: }
585: }

```



## Namespaces

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

## 1.2.7 FrmAnaSayfa.Designer.cs

This is file FrmAnaSayfa.Designer.cs.

## Body Source

```

1: ?namespace CSharpHelperMethods
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.pnlTarihIslemleri.SuspendLayout();
86:         this.pnlSifreIslemleri.SuspendLayout();
87:         this.pnlSayiIslemleri.SuspendLayout();
88:         this.pnlKisiIslemleri.SuspendLayout();
89:         this.pnlMetinIslemleri.SuspendLayout();
90:         this.SuspendLayout();
91:         //
92:         // cmbIslemTuru
93:         //
94:         this.cmbIslemTuru.FormattingEnabled = true;
95:         this.cmbIslemTuru.Items.AddRange(new object[] { "Metin Islemleri", "Tarih
Islemleri", "Sayi Islemleri", "Sifre Islemleri", "Kisi Islemleri" });
96:         this.cmbIslemTuru.Location = new System.Drawing.Point(132, 13);
97:         this.cmbIslemTuru.Name = "cmbIslemTuru";
98:         this.cmbIslemTuru.Size = new System.Drawing.Size(200, 21);
99:         this.cmbIslemTuru.TabIndex = 0;
100:        this.cmbIslemTuru.SelectedIndexChanged += new
System.EventHandler(this.cmbIslemTuru_SelectedIndexChanged);
101:        //
102:        // label1
103:        //
104:        this.label1.AutoSize = true;
105:        this.label1.Location = new System.Drawing.Point(12, 16);
106:        this.label1.Name = "label1";
107:        this.label1.Size = new System.Drawing.Size(93, 13);
108:        this.label1.TabIndex = 1;
109:        this.label1.Text = "Islem Türü Seçiniz";
110:        //
111:        // pnlTarihIslemleri
112:        //
113:        this.pnlTarihIslemleri.BorderStyle =
System.Windows.Forms.BorderStyle.FixedSingle;
114:        this.pnlTarihIslemleri.Controls.Add(this.btnAyFarki);
115:        this.pnlTarihIslemleri.Controls.Add(this.label8);
116:        this.pnlTarihIslemleri.Controls.Add(this.btnTarihAraligi);
117:        this.pnlTarihIslemleri.Controls.Add(this.btnYasHesaplaMetinsel);
118:        this.pnlTarihIslemleri.Controls.Add(this.btnYasHesapla);

```

```

119:         this.pnlTarihIslemleri.Controls.Add(this.label4);
120:         this.pnlTarihIslemleri.Controls.Add(this.lbSonuc);
121:         this.pnlTarihIslemleri.Controls.Add(this.dtpBitisTarihi);
122:         this.pnlTarihIslemleri.Controls.Add(this.dtpBaslangicTarihi);
123:         this.pnlTarihIslemleri.Controls.Add(this.label3);
124:         this.pnlTarihIslemleri.Controls.Add(this.label2);
125:         this.pnlTarihIslemleri.Location = new System.Drawing.Point(15, 56);
126:         this.pnlTarihIslemleri.Name = "pnlTarihIslemleri";
127:         this.pnlTarihIslemleri.Size = new System.Drawing.Size(374, 181);
128:         this.pnlTarihIslemleri.TabIndex = 2;
129:         //
130:         // btnAyFarki
131:         //
132:         this.btnAyFarki.Location = new System.Drawing.Point(286, 76);
133:         this.btnAyFarki.Name = "btnAyFarki";
134:         this.btnAyFarki.Size = new System.Drawing.Size(79, 23);
135:         this.btnAyFarki.TabIndex = 12;
136:         this.btnAyFarki.Text = "Ay Farki";
137:         this.btnAyFarki.UseVisualStyleBackColor = true;
138:         this.btnAyFarki.Click += new System.EventHandler(this.btnAyFarki_Click);
139:         //
140:         // label8
141:         //
142:         this.label8.AutoSize = true;
143:         this.label8.Dock = System.Windows.Forms.DockStyle.Top;
144:         this.label8.Location = new System.Drawing.Point(0, 0);
145:         this.label8.Name = "label8";
146:         this.label8.Size = new System.Drawing.Size(71, 13);
147:         this.label8.TabIndex = 11;
148:         this.label8.Text = "Tarih Islemleri";
149:         //
150:         // btnTarihAraligi
151:         //
152:         this.btnTarihAraligi.Location = new System.Drawing.Point(204, 76);
153:         this.btnTarihAraligi.Name = "btnTarihAraligi";
154:         this.btnTarihAraligi.Size = new System.Drawing.Size(76, 23);
155:         this.btnTarihAraligi.TabIndex = 10;
156:         this.btnTarihAraligi.Text = "Tarih Araligi";
157:         this.btnTarihAraligi.UseVisualStyleBackColor = true;
158:         this.btnTarihAraligi.Click += new
System.EventHandler(this.btnTarihAraligi_Click);
159:         //
160:         // btnYasHesaplaMetinsel
161:         //
162:         this.btnYasHesaplaMetinsel.Location = new System.Drawing.Point(84, 76);
163:         this.btnYasHesaplaMetinsel.Name = "btnYasHesaplaMetinsel";
164:         this.btnYasHesaplaMetinsel.Size = new System.Drawing.Size(113, 23);
165:         this.btnYasHesaplaMetinsel.TabIndex = 9;
166:         this.btnYasHesaplaMetinsel.Text = "Yas Hesapla (Metin)";
167:         this.btnYasHesaplaMetinsel.UseVisualStyleBackColor = true;
168:         this.btnYasHesaplaMetinsel.Click += new
System.EventHandler(this.btnYasHesaplaMetinsel_Click);
169:         //
170:         // btnYasHesapla
171:         //
172:         this.btnYasHesapla.Location = new System.Drawing.Point(3, 76);
173:         this.btnYasHesapla.Name = "btnYasHesapla";
174:         this.btnYasHesapla.Size = new System.Drawing.Size(75, 23);
175:         this.btnYasHesapla.TabIndex = 8;
176:         this.btnYasHesapla.Text = "Yas Hesapla";
177:         this.btnYasHesapla.UseVisualStyleBackColor = true;
178:         this.btnYasHesapla.Click += new
System.EventHandler(this.btnYasHesapla_Click);
179:         //
180:         // label4
181:         //
182:         this.label4.AutoSize = true;
183:         this.label4.Location = new System.Drawing.Point(18, 109);
184:         this.label4.Name = "label4";

```

```

185:         this.label4.Size = new System.Drawing.Size(38, 13);
186:         this.label4.TabIndex = 7;
187:         this.label4.Text = "Sonuç";
188:         //
189:         // lbSonuc
190:         //
191:         this.lbSonuc.FormattingEnabled = true;
192:         this.lbSonuc.HorizontalScrollbar = true;
193:         this.lbSonuc.Location = new System.Drawing.Point(62, 105);
194:         this.lbSonuc.Name = "lbSonuc";
195:         this.lbSonuc.Size = new System.Drawing.Size(256, 69);
196:         this.lbSonuc.TabIndex = 6;
197:         //
198:         // dtpBitisTarihi
199:         //
200:         this.dtpBitisTarihi.Format =
System.Windows.Forms.DateTimePickerFormat.Short;
201:         this.dtpBitisTarihi.Location = new System.Drawing.Point(117, 43);
202:         this.dtpBitisTarihi.Name = "dtpBitisTarihi";
203:         this.dtpBitisTarihi.ShowCheckBox = true;
204:         this.dtpBitisTarihi.Size = new System.Drawing.Size(200, 20);
205:         this.dtpBitisTarihi.TabIndex = 5;
206:         //
207:         // dtpBaslangicTarihi
208:         //
209:         this.dtpBaslangicTarihi.Format =
System.Windows.Forms.DateTimePickerFormat.Short;
210:         this.dtpBaslangicTarihi.Location = new System.Drawing.Point(117, 17);
211:         this.dtpBaslangicTarihi.Name = "dtpBaslangicTarihi";
212:         this.dtpBaslangicTarihi.ShowCheckBox = true;
213:         this.dtpBaslangicTarihi.Size = new System.Drawing.Size(200, 20);
214:         this.dtpBaslangicTarihi.TabIndex = 4;
215:         //
216:         // label3
217:         //
218:         this.label3.AutoSize = true;
219:         this.label3.Location = new System.Drawing.Point(44, 47);
220:         this.label3.Name = "label3";
221:         this.label3.Size = new System.Drawing.Size(55, 13);
222:         this.label3.TabIndex = 3;
223:         this.label3.Text = "Bitis Tarihi";
224:         //
225:         // label2
226:         //
227:         this.label2.AutoSize = true;
228:         this.label2.Location = new System.Drawing.Point(17, 24);
229:         this.label2.Name = "label2";
230:         this.label2.Size = new System.Drawing.Size(82, 13);
231:         this.label2.TabIndex = 2;
232:         this.label2.Text = "Baslangic Tarihi";
233:         //
234:         // pnlSifreIslemleri
235:         //
236:         this.pnlSifreIslemleri.BorderStyle =
System.Windows.Forms.BorderStyle.FixedSingle;
237:         this.pnlSifreIslemleri.Controls.Add(this.label7);
238:         this.pnlSifreIslemleri.Controls.Add(this.btnSifreUygunMu);
239:         this.pnlSifreIslemleri.Controls.Add(this.btnRastgeleSifreUret);
240:         this.pnlSifreIslemleri.Controls.Add(this.btnSha512Hash);
241:         this.pnlSifreIslemleri.Controls.Add(this.btnSha256Hash);
242:         this.pnlSifreIslemleri.Controls.Add(this.label6);
243:         this.pnlSifreIslemleri.Controls.Add(this.lbSifreSonuc);
244:         this.pnlSifreIslemleri.Controls.Add(this.btnMd5Hash);
245:         this.pnlSifreIslemleri.Controls.Add(this.txtSifrelenecekMetin);
246:         this.pnlSifreIslemleri.Controls.Add(this.label5);
247:         this.pnlSifreIslemleri.Location = new System.Drawing.Point(411, 56);
248:         this.pnlSifreIslemleri.Name = "pnlSifreIslemleri";
249:         this.pnlSifreIslemleri.Size = new System.Drawing.Size(377, 181);
250:         this.pnlSifreIslemleri.TabIndex = 3;

```

```

251:         //
252:         // label7
253:         //
254:         this.label7.AutoSize = true;
255:         this.label7.Dock = System.Windows.Forms.DockStyle.Top;
256:         this.label7.Location = new System.Drawing.Point(0, 0);
257:         this.label7.Name = "label7";
258:         this.label7.Size = new System.Drawing.Size(68, 13);
259:         this.label7.TabIndex = 18;
260:         this.label7.Text = "Sifre Islemleri";
261:         this.label7.TextAlign = System.Drawing.ContentAlignment.TopCenter;
262:         //
263:         // btnSifreUygunMu
264:         //
265:         this.btnSifreUygunMu.Location = new System.Drawing.Point(173, 71);
266:         this.btnSifreUygunMu.Name = "btnSifreUygunMu";
267:         this.btnSifreUygunMu.Size = new System.Drawing.Size(145, 23);
268:         this.btnSifreUygunMu.TabIndex = 17;
269:         this.btnSifreUygunMu.Text = "Sifre Uygun Mu?";
270:         this.btnSifreUygunMu.UseVisualStyleBackColor = true;
271:         this.btnSifreUygunMu.Click += new
System.EventHandler(this.btnSifreUygunMu_Click);
272:         //
273:         // btnRastgeleSifreUret
274:         //
275:         this.btnRastgeleSifreUret.Location = new System.Drawing.Point(21, 71);
276:         this.btnRastgeleSifreUret.Name = "btnRastgeleSifreUret";
277:         this.btnRastgeleSifreUret.Size = new System.Drawing.Size(145, 23);
278:         this.btnRastgeleSifreUret.TabIndex = 16;
279:         this.btnRastgeleSifreUret.Text = "Rastgele Sifre Üret";
280:         this.btnRastgeleSifreUret.UseVisualStyleBackColor = true;
281:         this.btnRastgeleSifreUret.Click += new
System.EventHandler(this.btnRastgeleSifreUret_Click);
282:         //
283:         // btnSha512Hash
284:         //
285:         this.btnSha512Hash.Location = new System.Drawing.Point(243, 42);
286:         this.btnSha512Hash.Name = "btnSha512Hash";
287:         this.btnSha512Hash.Size = new System.Drawing.Size(75, 23);
288:         this.btnSha512Hash.TabIndex = 15;
289:         this.btnSha512Hash.Text = "SHA 512";
290:         this.btnSha512Hash.UseVisualStyleBackColor = true;
291:         this.btnSha512Hash.Click += new
System.EventHandler(this.btnSha512Hash_Click);
292:         //
293:         // btnSha256Hash
294:         //
295:         this.btnSha256Hash.Location = new System.Drawing.Point(131, 42);
296:         this.btnSha256Hash.Name = "btnSha256Hash";
297:         this.btnSha256Hash.Size = new System.Drawing.Size(75, 23);
298:         this.btnSha256Hash.TabIndex = 14;
299:         this.btnSha256Hash.Text = "SHA 256";
300:         this.btnSha256Hash.UseVisualStyleBackColor = true;
301:         this.btnSha256Hash.Click += new
System.EventHandler(this.btnSha256Hash_Click);
302:         //
303:         // label6
304:         //
305:         this.label6.AutoSize = true;
306:         this.label6.Location = new System.Drawing.Point(21, 100);
307:         this.label6.Name = "label6";
308:         this.label6.Size = new System.Drawing.Size(38, 13);
309:         this.label6.TabIndex = 13;
310:         this.label6.Text = "Sonuç";
311:         //
312:         // lbSifreSonuc
313:         //
314:         this.lbSifreSonuc.FormattingEnabled = true;
315:         this.lbSifreSonuc.HorizontalScrollbar = true;

```



```

316:         this.lbSifreSonuc.Location = new System.Drawing.Point(65, 100);
317:         this.lbSifreSonuc.Name = "lbSifreSonuc";
318:         this.lbSifreSonuc.Size = new System.Drawing.Size(251, 69);
319:         this.lbSifreSonuc.TabIndex = 12;
320:         //
321:         // btnMd5Hash
322:         //
323:         this.btnMd5Hash.Location = new System.Drawing.Point(21, 42);
324:         this.btnMd5Hash.Name = "btnMd5Hash";
325:         this.btnMd5Hash.Size = new System.Drawing.Size(75, 23);
326:         this.btnMd5Hash.TabIndex = 11;
327:         this.btnMd5Hash.Text = "MD5";
328:         this.btnMd5Hash.UseVisualStyleBackColor = true;
329:         this.btnMd5Hash.Click += new System.EventHandler(this.btnMd5Hash_Click);
330:         //
331:         // txtSifrelenecekMetin
332:         //
333:         this.txtSifrelenecekMetin.Location = new System.Drawing.Point(118, 16);
334:         this.txtSifrelenecekMetin.Name = "txtSifrelenecekMetin";
335:         this.txtSifrelenecekMetin.Size = new System.Drawing.Size(199, 20);
336:         this.txtSifrelenecekMetin.TabIndex = 4;
337:         //
338:         // label5
339:         //
340:         this.label5.AutoSize = true;
341:         this.label5.Location = new System.Drawing.Point(17, 19);
342:         this.label5.Name = "label5";
343:         this.label5.Size = new System.Drawing.Size(95, 13);
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(15, 257);
359:         this.pnlSayiIslemleri.Name = "pnlSayiIslemleri";
360:         this.pnlSayiIslemleri.Size = new System.Drawing.Size(377, 181);
361:         this.pnlSayiIslemleri.TabIndex = 19;
362:         //
363:         // label9
364:         //
365:         this.label9.AutoSize = true;
366:         this.label9.Dock = System.Windows.Forms.DockStyle.Top;
367:         this.label9.Location = new System.Drawing.Point(0, 0);
368:         this.label9.Name = "label9";
369:         this.label9.Size = new System.Drawing.Size(67, 13);
370:         this.label9.TabIndex = 18;
371:         this.label9.Text = "Sayi Islemleri";
372:         this.label9.TextAlign = System.Drawing.ContentAlignment.TopCenter;
373:         //
374:         // btnOnayKoduOlustur
375:         //
376:         this.btnOnayKoduOlustur.Location = new System.Drawing.Point(243, 42);
377:         this.btnOnayKoduOlustur.Name = "btnOnayKoduOlustur";
378:         this.btnOnayKoduOlustur.Size = new System.Drawing.Size(75, 23);
379:         this.btnOnayKoduOlustur.TabIndex = 15;
380:         this.btnOnayKoduOlustur.Text = "Onay Kodu";
381:         this.btnOnayKoduOlustur.UseVisualStyleBackColor = true;
382:         this.btnOnayKoduOlustur.Click += new
System.EventHandler(this.btnOnayKoduOlustur_Click);

```

```

383:         //
384:         // btnMetneDonustur
385:         //
386:         this.btnMetneDonustur.Location = new System.Drawing.Point(102, 42);
387:         this.btnMetneDonustur.Name = "btnMetneDonustur";
388:         this.btnMetneDonustur.Size = new System.Drawing.Size(135, 23);
389:         this.btnMetneDonustur.TabIndex = 14;
390:         this.btnMetneDonustur.Text = "Para Metnine Dönüştür";
391:         this.btnMetneDonustur.UseVisualStyleBackColor = true;
392:         this.btnMetneDonustur.Click += new
System.EventHandler(this.btnMetneDonustur_Click);
393:         //
394:         // label10
395:         //
396:         this.label10.AutoSize = true;
397:         this.label10.Location = new System.Drawing.Point(18, 78);
398:         this.label10.Name = "label10";
399:         this.label10.Size = new System.Drawing.Size(38, 13);
400:         this.label10.TabIndex = 13;
401:         this.label10.Text = "Sonuç";
402:         //
403:         // lbSayiSonuc
404:         //
405:         this.lbSayiSonuc.FormattingEnabled = true;
406:         this.lbSayiSonuc.HorizontalScrollbar = true;
407:         this.lbSayiSonuc.Location = new System.Drawing.Point(62, 78);
408:         this.lbSayiSonuc.Name = "lbSayiSonuc";
409:         this.lbSayiSonuc.Size = new System.Drawing.Size(254, 82);
410:         this.lbSayiSonuc.TabIndex = 12;
411:         //
412:         // btnSayisalMi
413:         //
414:         this.btnSayisalMi.Location = new System.Drawing.Point(21, 42);
415:         this.btnSayisalMi.Name = "btnSayisalMi";
416:         this.btnSayisalMi.Size = new System.Drawing.Size(75, 23);
417:         this.btnSayisalMi.TabIndex = 11;
418:         this.btnSayisalMi.Text = "Sayisal Mi?";
419:         this.btnSayisalMi.UseVisualStyleBackColor = true;
420:         this.btnSayisalMi.Click += new
System.EventHandler(this.btnSayisalMi_Click);
421:         //
422:         // txtSayi
423:         //
424:         this.txtSayi.Location = new System.Drawing.Point(118, 16);
425:         this.txtSayi.Name = "txtSayi";
426:         this.txtSayi.Size = new System.Drawing.Size(199, 20);
427:         this.txtSayi.TabIndex = 4;
428:         //
429:         // label11
430:         //
431:         this.label11.AutoSize = true;
432:         this.label11.Location = new System.Drawing.Point(17, 19);
433:         this.label11.Name = "label11";
434:         this.label11.Size = new System.Drawing.Size(95, 13);
435:         this.label11.TabIndex = 3;
436:         this.label11.Text = "Sifrelenecek Metin";
437:         //
438:         // pnlKisiIslemleri
439:         //
440:         this.pnlKisiIslemleri.BorderStyle =
System.Windows.Forms.BorderStyle.FixedSingle;
441:         this.pnlKisiIslemleri.Controls.Add(this.label12);
442:         this.pnlKisiIslemleri.Controls.Add(this.btnEPostaDogrula);
443:         this.pnlKisiIslemleri.Controls.Add(this.btnIbanDogrula);
444:         this.pnlKisiIslemleri.Controls.Add(this.label13);
445:         this.pnlKisiIslemleri.Controls.Add(this.lbKisiSonuc);
446:         this.pnlKisiIslemleri.Controls.Add(this.btnTcKimlikDogrula);
447:         this.pnlKisiIslemleri.Controls.Add(this.txtKisiMetni);
448:         this.pnlKisiIslemleri.Controls.Add(this.label14);

```

```

449:         this.pnlKisiIslemleri.Location = new System.Drawing.Point(411, 257);
450:         this.pnlKisiIslemleri.Name = "pnlKisiIslemleri";
451:         this.pnlKisiIslemleri.Size = new System.Drawing.Size(377, 181);
452:         this.pnlKisiIslemleri.TabIndex = 20;
453:         //
454:         // label12
455:         //
456:         this.label12.AutoSize = true;
457:         this.label12.Dock = System.Windows.Forms.DockStyle.Top;
458:         this.label12.Location = new System.Drawing.Point(0, 0);
459:         this.label12.Name = "label12";
460:         this.label12.Size = new System.Drawing.Size(63, 13);
461:         this.label12.TabIndex = 18;
462:         this.label12.Text = "Kisi Islemleri";
463:         this.label12.TextAlign = System.Drawing.ContentAlignment.TopCenter;
464:         //
465:         // btnEPostaDogrula
466:         //
467:         this.btnEPostaDogrula.Location = new System.Drawing.Point(223, 42);
468:         this.btnEPostaDogrula.Name = "btnEPostaDogrula";
469:         this.btnEPostaDogrula.Size = new System.Drawing.Size(95, 23);
470:         this.btnEPostaDogrula.TabIndex = 15;
471:         this.btnEPostaDogrula.Text = "E-Posta Dogrula";
472:         this.btnEPostaDogrula.UseVisualStyleBackColor = true;
473:         this.btnEPostaDogrula.Click += new
System.EventHandler(this.btnEPostaDogrula_Click);
474:         //
475:         // btnIbanDogrula
476:         //
477:         this.btnIbanDogrula.Location = new System.Drawing.Point(131, 42);
478:         this.btnIbanDogrula.Name = "btnIbanDogrula";
479:         this.btnIbanDogrula.Size = new System.Drawing.Size(86, 23);
480:         this.btnIbanDogrula.TabIndex = 14;
481:         this.btnIbanDogrula.Text = "IBAN Dogrula";
482:         this.btnIbanDogrula.UseVisualStyleBackColor = true;
483:         this.btnIbanDogrula.Click += new
System.EventHandler(this.btnIbanDogrula_Click);
484:         //
485:         // label13
486:         //
487:         this.label13.AutoSize = true;
488:         this.label13.Location = new System.Drawing.Point(18, 78);
489:         this.label13.Name = "label13";
490:         this.label13.Size = new System.Drawing.Size(38, 13);
491:         this.label13.TabIndex = 13;
492:         this.label13.Text = "Sonuç";
493:         //
494:         // lbKisiSonuc
495:         //
496:         this.lbKisiSonuc.FormattingEnabled = true;
497:         this.lbKisiSonuc.HorizontalScrollbar = true;
498:         this.lbKisiSonuc.Location = new System.Drawing.Point(62, 78);
499:         this.lbKisiSonuc.Name = "lbKisiSonuc";
500:         this.lbKisiSonuc.Size = new System.Drawing.Size(254, 82);
501:         this.lbKisiSonuc.TabIndex = 12;
502:         //
503:         // btnTcKimlikDogrula
504:         //
505:         this.btnTcKimlikDogrula.Location = new System.Drawing.Point(21, 42);
506:         this.btnTcKimlikDogrula.Name = "btnTcKimlikDogrula";
507:         this.btnTcKimlikDogrula.Size = new System.Drawing.Size(104, 23);
508:         this.btnTcKimlikDogrula.TabIndex = 11;
509:         this.btnTcKimlikDogrula.Text = "TC Kimlik Dogrula";
510:         this.btnTcKimlikDogrula.UseVisualStyleBackColor = true;
511:         this.btnTcKimlikDogrula.Click += new
System.EventHandler(this.btnTcKimlikDogrula_Click);
512:         //
513:         // txtKisiMetni
514:         //

```



```

515:         this.txtKisiMetni.Location = new System.Drawing.Point(118, 16);
516:         this.txtKisiMetni.Name = "txtKisiMetni";
517:         this.txtKisiMetni.Size = new System.Drawing.Size(199, 20);
518:         this.txtKisiMetni.TabIndex = 4;
519:         //
520:         // label14
521:         //
522:         this.label14.AutoSize = true;
523:         this.label14.Location = new System.Drawing.Point(17, 19);
524:         this.label14.Name = "label14";
525:         this.label14.Size = new System.Drawing.Size(33, 13);
526:         this.label14.TabIndex = 3;
527:         this.label14.Text = "Metin";
528:         //
529:         // pnlMetinIslemleri
530:         //
531:         this.pnlMetinIslemleri.BorderStyle =
System.Windows.Forms.BorderStyle.FixedSingle;
532:         this.pnlMetinIslemleri.Controls.Add(this.btnTelefonNoDuzenle);
533:         this.pnlMetinIslemleri.Controls.Add(this.btnMetinSifrele);
534:         this.pnlMetinIslemleri.Controls.Add(this.label15);
535:         this.pnlMetinIslemleri.Controls.Add(this.btnTurkceKarakterDuzelt);
536:         this.pnlMetinIslemleri.Controls.Add(this.btnIlkHarfiBuyukYap);
537:         this.pnlMetinIslemleri.Controls.Add(this.label16);
538:         this.pnlMetinIslemleri.Controls.Add(this.lbMetinSonuc);
539:         this.pnlMetinIslemleri.Controls.Add(this.btnIlkHarfleriBuyukYap);
540:         this.pnlMetinIslemleri.Controls.Add(this.txtMetin);
541:         this.pnlMetinIslemleri.Controls.Add(this.label17);
542:         this.pnlMetinIslemleri.Location = new System.Drawing.Point(15, 458);
543:         this.pnlMetinIslemleri.Name = "pnlMetinIslemleri";
544:         this.pnlMetinIslemleri.Size = new System.Drawing.Size(773, 181);
545:         this.pnlMetinIslemleri.TabIndex = 21;
546:         //
547:         // btnTelefonNoDuzenle
548:         //
549:         this.btnTelefonNoDuzenle.Location = new System.Drawing.Point(503, 42);
550:         this.btnTelefonNoDuzenle.Name = "btnTelefonNoDuzenle";
551:         this.btnTelefonNoDuzenle.Size = new System.Drawing.Size(134, 23);
552:         this.btnTelefonNoDuzenle.TabIndex = 20;
553:         this.btnTelefonNoDuzenle.Text = "Telefon No Düzenle";
554:         this.btnTelefonNoDuzenle.UseVisualStyleBackColor = true;
555:         this.btnTelefonNoDuzenle.Click += new
System.EventHandler(this.btnTelefonNoDuzenle_Click);
556:         //
557:         // btnMetinSifrele
558:         //
559:         this.btnMetinSifrele.Location = new System.Drawing.Point(363, 42);
560:         this.btnMetinSifrele.Name = "btnMetinSifrele";
561:         this.btnMetinSifrele.Size = new System.Drawing.Size(134, 23);
562:         this.btnMetinSifrele.TabIndex = 19;
563:         this.btnMetinSifrele.Text = "Metin Sifrele";
564:         this.btnMetinSifrele.UseVisualStyleBackColor = true;
565:         this.btnMetinSifrele.Click += new
System.EventHandler(this.btnMetinSifrele_Click);
566:         //
567:         // label15
568:         //
569:         this.label15.AutoSize = true;
570:         this.label15.Dock = System.Windows.Forms.DockStyle.Top;
571:         this.label15.Location = new System.Drawing.Point(0, 0);
572:         this.label15.Name = "label15";
573:         this.label15.Size = new System.Drawing.Size(73, 13);
574:         this.label15.TabIndex = 18;
575:         this.label15.Text = "Metin Islemleri";
576:         this.label15.TextAlign = System.Drawing.ContentAlignment.TopCenter;
577:         //
578:         // btnTurkceKarakterDuzelt
579:         //
580:         this.btnTurkceKarakterDuzelt.Location = new System.Drawing.Point(223, 42);

```

```

581:         this.btnTurkceKarakterDuzelt.Name = "btnTurkceKarakterDuzelt";
582:         this.btnTurkceKarakterDuzelt.Size = new System.Drawing.Size(134, 23);
583:         this.btnTurkceKarakterDuzelt.TabIndex = 15;
584:         this.btnTurkceKarakterDuzelt.Text = "Türkçe Karakter Düzelt";
585:         this.btnTurkceKarakterDuzelt.UseVisualStyleBackColor = true;
586:         this.btnTurkceKarakterDuzelt.Click += new
System.EventHandler(this.btnTurkceKarakterDuzelt_Click);
587:         //
588:         // btnIlkHarfiBuyukYap
589:         //
590:         this.btnIlkHarfiBuyukYap.Location = new System.Drawing.Point(131, 42);
591:         this.btnIlkHarfiBuyukYap.Name = "btnIlkHarfiBuyukYap";
592:         this.btnIlkHarfiBuyukYap.Size = new System.Drawing.Size(86, 23);
593:         this.btnIlkHarfiBuyukYap.TabIndex = 14;
594:         this.btnIlkHarfiBuyukYap.Text = "Ilk Harf Büyük";
595:         this.btnIlkHarfiBuyukYap.UseVisualStyleBackColor = true;
596:         this.btnIlkHarfiBuyukYap.Click += new
System.EventHandler(this.btnIlkHarfiBuyukYap_Click);
597:         //
598:         // label16
599:         //
600:         this.label16.AutoSize = true;
601:         this.label16.Location = new System.Drawing.Point(18, 78);
602:         this.label16.Name = "label16";
603:         this.label16.Size = new System.Drawing.Size(38, 13);
604:         this.label16.TabIndex = 13;
605:         this.label16.Text = "Sonuç";
606:         //
607:         // lbMetinSonuc
608:         //
609:         this.lbMetinSonuc.FormattingEnabled = true;
610:         this.lbMetinSonuc.HorizontalScrollbar = true;
611:         this.lbMetinSonuc.Location = new System.Drawing.Point(62, 78);
612:         this.lbMetinSonuc.Name = "lbMetinSonuc";
613:         this.lbMetinSonuc.Size = new System.Drawing.Size(652, 82);
614:         this.lbMetinSonuc.TabIndex = 12;
615:         //
616:         // btnIlkHarfleriBuyukYap
617:         //
618:         this.btnIlkHarfleriBuyukYap.Location = new System.Drawing.Point(21, 42);
619:         this.btnIlkHarfleriBuyukYap.Name = "btnIlkHarfleriBuyukYap";
620:         this.btnIlkHarfleriBuyukYap.Size = new System.Drawing.Size(104, 23);
621:         this.btnIlkHarfleriBuyukYap.TabIndex = 11;
622:         this.btnIlkHarfleriBuyukYap.Text = "Ilk Harfler Büyük";
623:         this.btnIlkHarfleriBuyukYap.UseVisualStyleBackColor = true;
624:         this.btnIlkHarfleriBuyukYap.Click += new
System.EventHandler(this.btnIlkHarfleriBuyukYap_Click);
625:         //
626:         // txtMetin
627:         //
628:         this.txtMetin.Location = new System.Drawing.Point(119, 12);
629:         this.txtMetin.Name = "txtMetin";
630:         this.txtMetin.Size = new System.Drawing.Size(595, 20);
631:         this.txtMetin.TabIndex = 4;
632:         //
633:         // label17
634:         //
635:         this.label17.AutoSize = true;
636:         this.label17.Location = new System.Drawing.Point(17, 19);
637:         this.label17.Name = "label17";
638:         this.label17.Size = new System.Drawing.Size(33, 13);
639:         this.label17.TabIndex = 3;
640:         this.label17.Text = "Metin";
641:         //
642:         // FrmAnaSayfa
643:         //
644:         this.AutoScaleDimensions = new System.Drawing.SizeF(6F, 13F);
645:         this.AutoScaleMode = System.Windows.Forms.AutoScaleMode.Font;
646:         this.AutoSizeMode = System.Windows.Forms.AutoSizeMode.GrowAndShrink;

```

```

647:         this.ClientSize = new System.Drawing.Size(800, 686);
648:         this.Controls.Add(this.pnlMetinIslemleri);
649:         this.Controls.Add(this.pnlKisiIslemleri);
650:         this.Controls.Add(this.pnlSayiIslemleri);
651:         this.Controls.Add(this.pnlSifreIslemleri);
652:         this.Controls.Add(this.pnlTarihIslemleri);
653:         this.Controls.Add(this.label1);
654:         this.Controls.Add(this.cmbIslemTuru);
655:         this.Name = "FrmAnaSayfa";
656:         this.StartPosition = System.Windows.Forms.FormStartPosition.CenterScreen;
657:         this.Text = "Islemler Ekrani";
658:         this.Load += new System.EventHandler(this.FrmAnaSayfa_Load);
659:         this.pnlTarihIslemleri.ResumeLayout(false);
660:         this.pnlTarihIslemleri.PerformLayout();
661:         this.pnlSifreIslemleri.ResumeLayout(false);
662:         this.pnlSifreIslemleri.PerformLayout();
663:         this.pnlSayiIslemleri.ResumeLayout(false);
664:         this.pnlSayiIslemleri.PerformLayout();
665:         this.pnlKisiIslemleri.ResumeLayout(false);
666:         this.pnlKisiIslemleri.PerformLayout();
667:         this.pnlMetinIslemleri.ResumeLayout(false);
668:         this.pnlMetinIslemleri.PerformLayout();
669:         this.ResumeLayout(false);
670:         this.PerformLayout();
671:     }
672:
673:     #endregion
674:
675:     private System.Windows.Forms.ComboBox cmbIslemTuru;
676:     private System.Windows.Forms.Label label1;
677:     private System.Windows.Forms.Panel pnlTarihIslemleri;
678:     private System.Windows.Forms.Label label2;
679:     private System.Windows.Forms.Label label3;
680:     private System.Windows.Forms.DateTimePicker dtpBaslangicTarihi;
681:     private System.Windows.Forms.DateTimePicker dtpBitisTarihi;
682:     private System.Windows.Forms.ListBox lbSonuc;
683:     private System.Windows.Forms.Label label4;
684:     private System.Windows.Forms.Button btnYasHesapla;
685:     private System.Windows.Forms.Button btnYasHesaplaMetinsel;
686:     private System.Windows.Forms.Button btnTarihAraligi;
687:     private System.Windows.Forms.Panel pnlSifreIslemleri;
688:     private System.Windows.Forms.Label label5;
689:     private System.Windows.Forms.TextBox txtSifrelenecekMetin;
690:     private System.Windows.Forms.Button btnMd5Hash;
691:     private System.Windows.Forms.Label label6;
692:     private System.Windows.Forms.ListBox lbSifreSonuc;
693:     private System.Windows.Forms.Button btnSha256Hash;
694:     private System.Windows.Forms.Button btnSha512Hash;
695:     private System.Windows.Forms.Button btnRastgeleSifreUret;
696:     private System.Windows.Forms.Button btnSifreUygunMu;
697:     private System.Windows.Forms.Label label8;
698:     private System.Windows.Forms.Label label7;
699:     private System.Windows.Forms.Button btnAyFarki;
700:     private System.Windows.Forms.Panel pnlSayiIslemleri;
701:     private System.Windows.Forms.Label label9;
702:     private System.Windows.Forms.Button btnOnayKoduOlustur;
703:     private System.Windows.Forms.Button btnMetneDonustur;
704:     private System.Windows.Forms.Label label10;
705:     private System.Windows.Forms.ListBox lbSayiSonuc;
706:     private System.Windows.Forms.Button btnSayisalMi;
707:     private System.Windows.Forms.TextBox txtSayi;
708:     private System.Windows.Forms.Label label11;
709:     private System.Windows.Forms.Panel pnlKisiIslemleri;
710:     private System.Windows.Forms.Label label12;
711:     private System.Windows.Forms.Button btnEPostaDogrula;
712:     private System.Windows.Forms.Button btnIbanDogrula;
713:     private System.Windows.Forms.Label label13;
714:     private System.Windows.Forms.ListBox lbKisiSonuc;
715:     private System.Windows.Forms.Button btnTcKimlikDogrula;

```

```

716:         private System.Windows.Forms.TextBox txtKisiMetni;
717:         private System.Windows.Forms.Label label14;
718:         private System.Windows.Forms.Panel pnlMetinIslemleri;
719:         private System.Windows.Forms.Label label15;
720:         private System.Windows.Forms.Button btnTurkceKarakterDuzelt;
721:         private System.Windows.Forms.Button btnIlkHarfiBuyukYap;
722:         private System.Windows.Forms.Label label16;
723:         private System.Windows.Forms.ListBox lbMetinSonuc;
724:         private System.Windows.Forms.Button btnIlkHarfleriBuyukYap;
725:         private System.Windows.Forms.TextBox txtMetin;
726:         private System.Windows.Forms.Label label17;
727:         private System.Windows.Forms.Button btnMetinSifrele;
728:         private System.Windows.Forms.Button btnTelefonNoDuzenle;
729:     }
730: }
731:

```

### Namespaces

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

## 1.2.8 Kisilslemleri.cs

This is file Kisilslemleri.cs.

### Body Source

```

1: ?using System;
2: using System.Text;
3: using System.Text.RegularExpressions;
4:
5: namespace CSharpHelperMethods.YardimciSiniflar
6: {
7:     /// <summary>
8:     /// Kisi ile ilgili yardimci metotların tamamını içeren sınıf
9:     /// </summary>
10:    public static class KisiIslemleri
11:    {
12:        /// <summary>
13:        /// Girilen TC Kimlik Numarasının, standart TC Kimlik No algoritmasına uygun
14:        /// olup olmadığını kontrol eden metottur.
15:        /// </summary>
16:        /// <param name="tcKimlikNo">TC Kimlik No</param>
17:        /// <returns>TC Kimlik No'nun Uygun Olup Olmadığı 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) %
33:            10;
34:            var q2 = (10 - ((c2 + c4 + c6 + c8 + q1) * 3 + c1 + c3 + c5 + c7 + c9) %
35:            10) % 10;
36:            var dogrula = btcno * 100 + q1 * 10 + q2 == tcNo;
37:        }
38:    }
39: }

```

```

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
77:     /// olmadigini
78:     /// Kontrol eden metottur
79:     /// </summary>
80:     /// <param name="ePosta">E-Posta Adresi</param>
81:     /// <returns>E-Posta Adresinin Geçerli Olup Olmadigi Bilgisi</returns>
82:     public static bool DogrulaEPosta(string ePosta)
83:     {
84:         const string pattern = @"^(([\w-]+\.)+[\w-]+|([a-zA-Z]{1}|[\w-]{2,}))@"
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.

## 1.2.9 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.Text;
6: using System.Text.RegularExpressions;
7:
8: namespace CSharpHelperMethods.YardimciSiniflar
9: {
10:     /// <summary>
11:     /// Metinsel veri kontrol metotlarini barindiran sinif
12:     /// </summary>
13:     public static class MetinIslemleri
14:     {
15:         /// <summary>
16:         /// Girilen metnin basinda virgöl olup olmadigini kontrol eden metottur.
17:         /// </summary>
18:         /// <param name="text">Girdi metni</param>
19:         /// <returns>Girilen Metnin Olmasi Gerektigi Hali</returns>
20:         public static string IdealMetinGetir(string text)
21:         {
22:             if (!string.IsNullOrEmpty(text) && text[0] == ',')
23:             {
24:                 text = text.Substring(1, text.Length - 1);
25:             }
26:             return text;
27:         }
28:
29:         public static string MetinZararliKarakterDuzelt(string text)
30:         {
31:             text = text.Replace("&#252;", "ü");
32:             return text;
33:         }
34:
35:         public static string MetinSifrele(string metin)
36:         {
37:             if (string.IsNullOrEmpty(metin)) return metin;
38:             var metinDuzenleyici = new StringBuilder(metin);
39:             switch (metin.Length)
40:             {
41:                 case 10:
42:                     metinDuzenleyici.Remove(2, 6);
43:                     metinDuzenleyici.Insert(2, "*****");
44:                     metin = metinDuzenleyici.ToString();
45:                     break;
46:                 case 11:
47:                     metinDuzenleyici.Remove(2, 7);
48:                     metinDuzenleyici.Insert(2, "*****");
49:                     metin = metinDuzenleyici.ToString();
50:                     break;
51:             }
52:             return metin;
53:         }
54:
55:         /// <summary>
56:         /// Girilen metnin basinda virgöl olup olmadigini kontrol eden metottur.
57:         /// </summary>
58:         /// <param name="text">Girdi metni</param>
59:         /// <returns>Girilen Metnin Olmasi Gerektigi Hali</returns>

```



```

60:         public static string TurkceKarakterleriDuzelt(string text)
61:         {
62:             text = text.Replace("ü", "u");
63:             text = text.Replace("i", "i");
64:             text = text.Replace("ö", "o");
65:             text = text.Replace("ü", "u");
66:             text = text.Replace("s", "s");
67:             text = text.Replace("g", "g");
68:             text = text.Replace("ç", "c");
69:             text = text.Replace("Ü", "U");
70:             text = text.Replace("İ", "I");
71:             text = text.Replace("Ö", "O");
72:             text = text.Replace("Û", "U");
73:             text = text.Replace("S", "S");
74:             text = text.Replace("G", "G");
75:             text = text.Replace("Ç", "C");
76:             return text;
77:         }
78:
79:         /// <summary>
80:         /// Girilen metnin basinda virgöl olup olmadigini kontrol eden metottur.
81:         /// </summary>
82:         /// <param name="text">Girdi metni</param>
83:         /// <returns>Girilen Metnin Olmasi Gerektigi Hali</returns>
84:         public static string UTF8Duzelt(string text)
85:         {
86:             text = text.Replace("&ccedil;", "ç");
87:             text = text.Replace("&ouml;", "ö");
88:             text = text.Replace("&uuml;", "ü");
89:             text = text.Replace("&cedil;", "ç");
90:             text = text.Replace("&ouml;", "ö");
91:             text = text.Replace("&uuml;", "ü");
92:             return text;
93:         }
94:
95:         /// <summary>
96:         /// Girilen metnin basinda virgöl olup olmadigini kontrol eden metottur.
97:         /// </summary>
98:         /// <param name="text">Girdi metni</param>
99:         /// <returns>Girilen Metnin Olmasi Gerektigi Hali</returns>
100:         public static string HTMLDuzelt(string text)
101:         {
102:             var sc = new List<string>
103:             {
104:                 // get rid of unnecessary tag spans (comments and title)
105:                 @"<!--(\w|\W)+?-->",
106:                 @"<title>(\w|\W)+?</title>",
107:                 // Get rid of classes and styles
108:                 @"\s?class=\w+",
109:                 @"\s+style='[^']*'",
110:                 // Get rid of unnecessary tags
111:                 @"<(meta|link|/?o:|/?style|/?div|/?st\d|/?head|/?html|body|/?body|/?span|!\[)\[>]*?>",
112:                 // Get rid of empty paragraph tags
113:                 @"(<[>]+>)+&nbsp;(</\w+>)+",
114:                 @"<a[>]*(<[>]+)</a>",
115:                 @"<p[>]*(<[>]+)</p>",
116:                 // remove bizarre v: element attached to <img> tag
117:                 @"\s+v:\w+=\""[^"]*"\""",
118:                 // remove extra lines
119:                 @"(\n\r){2,}",
120:                 "&nbsp;"
121:             };
122:             text = sc.Aggregate(text, (current, s) => Regex.Replace(current, s,
123: 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;", "'");

```

```

126:         text = text.Replace("rsquo;", "'");
127:         text = text.Replace("&rdquo;", "\"");
128:         text = text.Replace("rdquo;", "\"");
129:
130:         text = text.Replace("&lsquo;", "'");
131:         text = text.Replace("lsquo;", "'");
132:         text = text.Replace("&ldquo;", "\"");
133:         text = text.Replace("ldquo;", "\"");
134:
135:         text = text.Replace("&lt;", "<");
136:         text = text.Replace("&gt;", ">");
137:         text = text.Replace("&amp;", "&");
138:         //text = text.Replace("&lt;", "");
139:         //text = text.Replace("&p&gt;", "");
140:         //text = text.Replace("/&p&gt;", "");
141:         //text = text.Replace("&gt;", "");
142:         //return text;
143:         return text;
144:     }
145:
146:     public static string MetinIlkKarakterBuyukDigerleriKucukYap(string text) =>
text.First().ToString().ToUpper() + text.Substring(1);
147:
148:     /// <summary>
149:     /// Girilen bir metindeki tüm kelimelerin ilk harflerinin
150:     /// Büyük yapilmasi için hazirlanan metottur
151:     /// </summary>
152:     /// <param name="input">Girilen metin</param>
153:     /// <returns>Metnin Kelimelerinin İlk Harflerinin Büyüültülmüs Hali</returns>
154:     public static string MetinIlkKarakterleriBuyukYap(this string input)
155:     {
156:         var result = "";
157:         var wordList = input.Split(' ');
158:         var cultureInfo = new CultureInfo("tr-TR");
159:         foreach (var item in wordList)
160:         {
161:             if (item.Length > 1)
162:                 result = result + " " + item.Substring(0, 1).ToUpper(cultureInfo) +
item.Substring(1).ToLower(cultureInfo) + " ";
163:             else if (item.Length == 1)
164:                 result = result + " " + item.ToUpper(cultureInfo) + " ";
165:         }
166:         return result;
167:     }
168:
169:     /// <summary>
170:     /// Girilen telefon numarasinin SMS gönderilecek hale getirilmesini saglayan
metottur.
171:     /// </summary>
172:     /// <param name="telefonNo">Telefon No Bilgisi</param>
173:     /// <returns>Girilen Telefon Numarasinin SMS Gönderilecek Hali</returns>
174:     public static string TelefonNoDuzenle(string telefonNo)
175:     {
176:         if (string.IsNullOrEmpty(telefonNo)) return string.Empty;
177:         telefonNo = telefonNo.Replace("-", "").Replace(" ", "").Replace("(",
"").Replace(")", "");
178:         if (telefonNo.Length == 11)
179:         {
180:             telefonNo = telefonNo.Substring(telefonNo.Length - 10);
181:         }
182:         return telefonNo;
183:     }
184:
185:
186:     /// <summary>
187:     /// Girilen metnin bosluklarinin kaldırilmasini saglayan metottur.
188:     /// </summary>
189:     /// <param name="metin">Telefon No Bilgisi</param>
190:     /// <returns>Girilen Metnin Istene Hali</returns>

```



```

191:         public static string BoslukKaldir(string metin)
192:         {
193:             if (string.IsNullOrEmpty(metin)) return string.Empty;
194:             metin = metin.Replace("-", "").Replace(" ", "").Replace("(",
195:             "").Replace(")", "");
196:             return metin;
197:         }
198:     }
199: }

```

#### Namespaces

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

## 1.2.10 Program.cs

This is file Program.cs.

#### Body Source

```

1: ?using System;
2: using System.Windows.Forms;
3:
4: namespace CSharpHelperMethods
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.

## 1.2.11 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:
23: [global::System.CodeDom.Compiler.GeneratedCodeAttribute("System.Resources.Tools.StronglyTypedResourceBuilder",
24: "4.0.0.0")]
25: [global::System.Diagnostics.DebuggerNonUserCodeAttribute()]
26: [global::System.Runtime.CompilerServices.CompilerGeneratedAttribute()]
27:     internal class Resources
28:     {
29:
30:         private static global::System.Resources.ResourceManager resourceMan;
31:
32:         private static global::System.Globalization.CultureInfo resourceCulture;
33:
34: [global::System.Diagnostics.CodeAnalysis.SuppressMessageAttribute("Microsoft.Performance",
35: "CA1811:AvoidUncalledPrivateCode")]
36:         internal Resources()
37:         {
38:
39:             /// <summary>
40:             ///     Returns the cached ResourceManager instance used by this class.
41:             /// </summary>
42:
43: [global::System.ComponentModel.EditorBrowsableAttribute(global::System.ComponentModel.EditorBrowsableState.Advanced)]
44:         internal static global::System.Resources.ResourceManager ResourceManager
45:         {
46:             get
47:             {
48:                 if ((resourceMan == null))
49:                 {
50:                     global::System.Resources.ResourceManager temp = new
51: global::System.Resources.ResourceManager("CSharpHelperMethods.Properties.Resources",
52: typeof(Resources).Assembly);
53:                     resourceMan = temp;
54:                 }
55:                 return resourceMan;
56:             }
57:
58:             /// <summary>
59:             ///     Overrides the current thread's CurrentUICulture property for all
60:             ///     resource lookups using this strongly typed resource class.
61:             /// </summary>
62:
63: [global::System.ComponentModel.EditorBrowsableAttribute(global::System.ComponentModel.EditorBrowsableState.Advanced)]
64:         internal static global::System.Globalization.CultureInfo Culture
65:         {
66:             get
67:             {
68:                 return resourceCulture;
69:             }
70:         }
71:     }

```

```

65:         set
66:         {
67:             resourceCulture = value;
68:         }
69:     }
70: }
71: }

```

### Namespaces

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

## 1.2.12 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.YardimciSiniflar
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 alınan string degerde virgül olduğundan 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 SayiHaliCevir(string metin)
28:        {
29:            decimal val;
30:            if (metin.Contains("_") || metin.Contains(".") || metin.Contains(","))
31:            {
32:                if (!decimal.TryParse(metin.Replace(",", "").Replace(".", "",
33:                " ").Replace("_", " "), NumberStyles.Number,
34:                CultureInfo.InvariantCulture, out val))
35:                {
36:                    return 0;
37:                }
38:                return val / 100;
39:            }
40:            var value = decimal.TryParse(metin.Replace(",", "").Replace(".", "",
41:            " ").Replace("_", " "), NumberStyles.Number,
42:            CultureInfo.InvariantCulture, out val);
43:            return value ? val : 0;
44:        }
45:
46:        /// <summary>
47:        /// Elimizdeki tutar verilerinin

```

```

45:         /// Yazı formatında gösterilmesini sağlayacak bir metottur
46:         /// </summary>
47:         /// <param name="tutar">Girilen Sayı Bilgisi</param>
48:         /// <returns>Sayının Yazıya Dönüştürülmüş Hali</returns>
49:         public static string SayıYazıyaCevir(decimal tutar)
50:         {
51:             var sTutar = tutar.ToString("F2").Replace('.', ','); // Replace('.', ',')
ondalik ayracının . olma durumu için
52:             var lira = sTutar.Substring(0, sTutar.IndexOf(',')); //tutarın tam kısmı
53:             var kurus = sTutar.Substring(sTutar.IndexOf(',') + 1, 2);
54:             var yazi = new StringBuilder();
55:
56:             var birler = new[] { "", "BİR", "İKİ", "ÜÇ", "DÖRT", "BES", "ALTI", "YEDİ",
"SEKİZ", "DOKUZ" };
57:             var onlar = new[] { "", "ON", "YIRMI", "OTUZ", "KIRK", "ELLİ", "ALTMIS",
"YETMİS", "SEKSEN", "DOKSAN" };
58:             var binler = new[] { "KENTRİLYON", "KATRİLYON", "TRİLYON", "MİLYAR",
"MİLYON", "BİN", "" };
59:
60:             const int grupSayisi = 7;
61:             //sayıdaki 3'lü grup sayısı. katrilyon içi 6. (1.234,00 daki grup sayısı
2'dir.)
62:             //KATRİLYON'un basına ekleyeceğiniz her değer için grup sayisini artiriniz.
63:
64:             lira = lira.PadLeft(grupSayisi * 3, '0'); //sayının soluna '0' eklenerek
sayı 'grup sayısı x 3' basakmakli yapiliyor.
65:
66:             for (var i = 0; i < grupSayisi * 3; i += 3) //sayı 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("BİR YÜZ"))
73:                     grupDegeri.Replace("BİR YÜZ", "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("BİR BİN"))
79:                     grupDegeri.Replace("BİR BİN", "BİN");
80:
81:                 yazi.Append(grupDegeri);
82:             }
83:
84:             if (yazi.Length > 0)
85:                 yazi.Append(" TL ");
86:
87:             var yaziUzunlugu = yazi.Length;
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 oluşan 6 haneli random sayı 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.

### 1.2.13 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.SettingsDesigner.SettingsSingleFileGenerator",
17: "11.0.0.0")]
18:     internal sealed partial class Settings :
19:         global::System.Configuration.ApplicationSettingsBase
20:     {
21:
22:         private static Settings defaultInstance =
23:             ((Settings)(global::System.Configuration.ApplicationSettingsBase.Synchronized(new
24: Settings())));
25:
26:         public static Settings Default
27:         {
28:             get
29:             {
30:                 return defaultInstance;
31:             }
32:         }
33:     }
34: }
```

## Namespaces

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

## 1.2.14 Sifrelemelslemleri.cs

This is file Sifrelemelslemleri.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.YardimciSiniflar
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) =>
25:                 $"{{current}}{{b:x2}}");
26:             return strHex.ToUpper();
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:         /// <summary>
46:         /// Kullanici sifresinin MD5 yöntemiyle sifrelenmesini saglayan metottur.
47:         /// </summary>
48:         /// <param name="text">Sifrelenecek Metin</param>
49:         /// <returns>Girdi Metninin Sifrelenmis Hali</returns>
50:         public static string HesaplaMD5(string text)
51:         {
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 dönüş 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 =
"abcdefghijklmnopqrstuvwxyzABCDEFGHIJKLMNOPQRSTUVWXYZ1234567890";
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.

## 1.2.15 Siralamalslemleri.cs

This is file Siralamalslemleri.cs.

## Body Source

```

1: ?using System;
2: using System.Globalization;
3: using System.Text;
4: using static System.String;
5:
6: namespace CSharpHelperMethods.YardimciSiniflar
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.

## 1.2.16 Tarihslemleri.cs

This is file Tarihslemleri.cs.

## Body Source

```

1: ?using System;
2: using System.Text;
3:
4: namespace CSharpHelperMethods.YardimciSiniflar
5: {
6:     /// <summary>
7:     /// Tarihsel veri kontrol metotlarini barindiran sinif
8:     /// </summary>
9:     public static class TarihIslemleri
10:    {
11:        /// <summary>
12:        /// Filtre ekranlarinda baslangic 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 kullanılan 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:            var months = 0;
47:            for (var i = 1; i <= 12; i++)
48:            {
49:                if (pastYearDate.AddMonths(i) == now)
50:                {
51:                    months = i;
52:                    break;
53:                }
54:
55:                if (pastYearDate.AddMonths(i) < now) continue;
56:                months = i - 1;
57:                break;
58:            }
59:            return years;
60:        }
61:
62:        /// <summary>
63:        /// Yas hesaplamak için kullanılan metottur

```

```

64:         /// </summary>
65:         /// <param name="dogumTarihi">Dogum Tarihi Bilgisi</param>
66:         /// <returns>Yas Bilgisi (Metinsel olarak)</returns>
67:         public static string YasHesaplaMetinsel(DateTime dogumTarihi)
68:         {
69:             var now = DateTime.Now;
70:             var years = new DateTime(DateTime.Now.Subtract(dogumTarihi).Ticks).Year -
1;
71:             var pastYearDate = dogumTarihi.AddYears(years);
72:             var months = 0;
73:             for (var i = 1; i <= 12; i++)
74:             {
75:                 if (pastYearDate.AddMonths(i) == now)
76:                 {
77:                     months = i;
78:                     break;
79:                 }
80:
81:                 if (pastYearDate.AddMonths(i) < now) continue;
82:                 months = i - 1;
83:                 break;
84:             }
85:             var days = now.Subtract(pastYearDate.AddMonths(months)).Days;
86:             var hours = now.Subtract(pastYearDate).Hours;
87:             var minutes = now.Subtract(pastYearDate).Minutes;
88:             var seconds = now.Subtract(pastYearDate).Seconds;
89:             return $"Yas: {years} Yil {months} Ay {days} Gün";
90:             //return String.Format("Yas: {0} Yil {1} Ay {2} Gün {3} Saat {4} Saniye",
91:             //years, months, Days, Hours, Seconds);
92:         }
93:
94:         /// <summary>
95:         /// Iki tarih arasinda kaç gün olduğunu hesaplamak için kullanılan metottur
96:         /// </summary>
97:         /// <param name="baslangicTarihi">Baslangıç Tarihi Bilgisi</param>
98:         /// <param name="bitisTarihi">Bitis Tarihi Bilgisi</param>
99:         /// <returns>Tarih Araligi Bilgisi</returns>
100:         public static string TarihAraligiHesapla(DateTime baslangicTarihi, DateTime
bitisTarihi)
101:         {
102:             var years = new DateTime(bitisTarihi.Subtract(baslangicTarihi).Ticks).Year
- 1;
103:             var pastYearDate = baslangicTarihi.AddYears(years);
104:             var months = 0;
105:             for (var i = 1; i <= 12; i++)
106:             {
107:                 if (pastYearDate.AddMonths(i) == bitisTarihi)
108:                 {
109:                     months = i;
110:                     break;
111:                 }
112:
113:                 if (pastYearDate.AddMonths(i) < bitisTarihi) continue;
114:                 months = i - 1;
115:                 break;
116:             }
117:             var days = bitisTarihi.Subtract(pastYearDate.AddMonths(months)).Days;
118:             var result = new StringBuilder();
119:             result.Append(years.ToString().PadLeft(2, '0'));
120:             result.Append(" Yil, ");
121:             result.Append(months.ToString().PadLeft(2, '0'));
122:             result.Append(" Ay, ");
123:             result.Append(days.ToString().PadLeft(2, '0'));
124:             result.Append(" Gün");
125:             return result.ToString();
126:         }
127:
128:         /// <summary>
129:         /// Kidem Tazminatı Hesaplaması Yapilirken ayda kaç gün olursa olsun tamamında

```

```

calismissa
130:      /// 30 gün üzerinden hesaplanması gerektiği için
131:      /// Isten ayrilis ve is baslangic tarihleri arasinda kaç gün geçtiğini
belirtmek için
132:      /// Hesaplama yapılmasını sağlayan metottur.
133:      /// </summary>
134:      /// <param name="baslangicTarihi">Baslangic Tarihi Bilgisi</param>
135:      /// <param name="bitisTarihi">Bitis Tarihi Bilgisi</param>
136:      /// <returns>Gün Sayisi Bilgisi</returns>
137:      public static int KidemTazminatıTarihAralığıHesapla(DateTime? baslangicTarihi,
DateTime? bitisTarihi)
138:      {
139:          var artacakMi = true;
140:          var result = 0;
141:          var baslangicMi = false;
142:          var bitisMi = false;
143:          var aylikGunSayisi = 0;
144:          for (var date = (DateTime)baslangicTarihi; date <= (DateTime)bitisTarihi;
date = date.AddDays(1))
145:          {
146:              if (!artacakMi)
147:                  result++;
148:              else
149:                  artacakMi = false;
150:
151:              if (date.Day == 1)
152:                  baslangicMi = true;
153:              else if (date.Day == DateTime.DaysInMonth(date.Year, date.Month))
154:                  bitisMi = true;
155:
156:              if (baslangicMi)
157:                  aylikGunSayisi++;
158:              if (baslangicMi && bitisMi && aylikGunSayisi != 30)
159:              {
160:                  result = result - aylikGunSayisi + 30;
161:                  baslangicMi = false;
162:                  bitisMi = false;
163:                  aylikGunSayisi = 0;
164:              }
165:              else if (bitisMi)
166:              {
167:                  baslangicMi = false;
168:                  bitisMi = false;
169:                  aylikGunSayisi = 0;
170:              }
171:
172:          }
173:          if (baslangicTarihi.Value.Day > bitisTarihi.Value.Day)
174:              result--;
175:          return result;
176:      }
177:
178:      /// <summary>
179:      /// Girilen iki tarih arasinda kaç ay olduğunu bulan metottur
180:      /// </summary>
181:      /// <param name="lValue">İlk tarih bilgisi</param>
182:      /// <param name="rValue">Son tarih bilgisi</param>
183:      /// <returns>Ay Farki Bilgisi</returns>
184:      public static int AyFarki(DateTime lValue, DateTime rValue) =>
Math.Abs(lValue.Month - rValue.Month + 12 * (lValue.Year - rValue.Year));
185:  }
186: }

```

## Namespaces

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

# Index

## A

Alphanumeric enumeration member 25  
AssemblyInfo.cs 63

## C

CSharpHelperMethods 1  
CSharpHelperMethods namespace 1  
    Classes 32  
CSharpHelperMethods.csproj 64  
CSharpHelperMethods.FrmAnaSayfa 32  
CSharpHelperMethods.FrmAnaSayfa.btnAyFarki 34  
CSharpHelperMethods.FrmAnaSayfa.btnEPostaDogrula 34  
CSharpHelperMethods.FrmAnaSayfa.btnIbanDogrula 34  
CSharpHelperMethods.FrmAnaSayfa.btnIlkHarfiBuyukYap 35  
CSharpHelperMethods.FrmAnaSayfa.btnIlkHarfleriBuyukYap 35  
CSharpHelperMethods.FrmAnaSayfa.btnMd5Hash 35  
CSharpHelperMethods.FrmAnaSayfa.btnMetinSifrele 35  
CSharpHelperMethods.FrmAnaSayfa.btnMetneDonustur 35  
CSharpHelperMethods.FrmAnaSayfa.btnOnayKoduOlustur 35  
CSharpHelperMethods.FrmAnaSayfa.btnRastgeleSifreUret 35  
CSharpHelperMethods.FrmAnaSayfa.btnSayisalMi 36  
CSharpHelperMethods.FrmAnaSayfa.btnSha256Hash 36  
CSharpHelperMethods.FrmAnaSayfa.btnSha512Hash 36  
CSharpHelperMethods.FrmAnaSayfa.btnSifreUygunMu 36  
CSharpHelperMethods.FrmAnaSayfa.btnTarihAraligi 36  
CSharpHelperMethods.FrmAnaSayfa.btnTcKimlikDogrula 36  
CSharpHelperMethods.FrmAnaSayfa.btnTelefonNoDuzenle 37  
CSharpHelperMethods.FrmAnaSayfa.btnTurkceKarakterDuzelt 37  
CSharpHelperMethods.FrmAnaSayfa.btnYasHesapla 37  
CSharpHelperMethods.FrmAnaSayfa.btnYasHesaplaMetinsel 37  
CSharpHelperMethods.FrmAnaSayfa.cmbIslemTuru 37  
CSharpHelperMethods.FrmAnaSayfa.cmbIslemTuru\_SelectedIndexChanged 42  
CSharpHelperMethods.FrmAnaSayfa.components 37  
CSharpHelperMethods.FrmAnaSayfa.Dispose 51  
CSharpHelperMethods.FrmAnaSayfa.dtpBaslangicTarihi 37  
CSharpHelperMethods.FrmAnaSayfa.dtpBitisTarihi 38  
CSharpHelperMethods.FrmAnaSayfa.FrmAnaSayfa 34  
CSharpHelperMethods.FrmAnaSayfa.FrmAnaSayfa\_Load 51  
CSharpHelperMethods.FrmAnaSayfa.InitializeComponent 52  
CSharpHelperMethods.FrmAnaSayfa.label1 38  
CSharpHelperMethods.FrmAnaSayfa.label10 38  
CSharpHelperMethods.FrmAnaSayfa.label11 38  
CSharpHelperMethods.FrmAnaSayfa.label12 38  
CSharpHelperMethods.FrmAnaSayfa.label13 38  
CSharpHelperMethods.FrmAnaSayfa.label14 38  
CSharpHelperMethods.FrmAnaSayfa.label15 39  
CSharpHelperMethods.FrmAnaSayfa.label16 39  
CSharpHelperMethods.FrmAnaSayfa.label17 39  
CSharpHelperMethods.FrmAnaSayfa.label2 39  
CSharpHelperMethods.FrmAnaSayfa.label3 39  
CSharpHelperMethods.FrmAnaSayfa.label4 39  
CSharpHelperMethods.FrmAnaSayfa.label5 40  
CSharpHelperMethods.FrmAnaSayfa.label6 40  
CSharpHelperMethods.FrmAnaSayfa.label7 40  
CSharpHelperMethods.FrmAnaSayfa.label8 40  
CSharpHelperMethods.FrmAnaSayfa.label9 40  
CSharpHelperMethods.FrmAnaSayfa.lbKisiSonuc 40  
CSharpHelperMethods.FrmAnaSayfa.lbMetinSonuc 40  
CSharpHelperMethods.FrmAnaSayfa.lbSayiSonuc 41  
CSharpHelperMethods.FrmAnaSayfa.lbSifreSonuc 41  
CSharpHelperMethods.FrmAnaSayfa.lbSonuc 41  
CSharpHelperMethods.FrmAnaSayfa.pnlKisilislemleri 41  
CSharpHelperMethods.FrmAnaSayfa.pnlMetinIslemleri 41  
CSharpHelperMethods.FrmAnaSayfa.pnlSayiIslemleri 41  
CSharpHelperMethods.FrmAnaSayfa.pnlSifreIslemleri 42  
CSharpHelperMethods.FrmAnaSayfa.pnlTarihIslemleri 42  
CSharpHelperMethods.FrmAnaSayfa.txtKisiMetni 42  
CSharpHelperMethods.FrmAnaSayfa.txtMetin 42  
CSharpHelperMethods.FrmAnaSayfa.txtSayi 42  
CSharpHelperMethods.FrmAnaSayfa.txtSifrelenecekMetin 42  
CSharpHelperMethods.Program 62  
CSharpHelperMethods.Program.Main 62  
CSharpHelperMethods.Properties 1  
CSharpHelperMethods.Properties namespace 1

Classes 1	10
CSharpHelperMethods.Properties.Resources 1	CSharpHelperMethods.YardimciSiniflar.KisiIslemleri 11
CSharpHelperMethods.Properties.Resources.Culture 3	CSharpHelperMethods.YardimciSiniflar.KisiIslemleri.DogrulaEPosta 11
CSharpHelperMethods.Properties.Resources.resourceCulture 2	CSharpHelperMethods.YardimciSiniflar.KisiIslemleri.DogrulaBAN 12
CSharpHelperMethods.Properties.Resources.resourceManager 3	CSharpHelperMethods.YardimciSiniflar.KisiIslemleri.DogrulaTcKimlikNo 12
CSharpHelperMethods.Properties.Resources.Resources 2	CSharpHelperMethods.YardimciSiniflar.MetinIslemleri 13
CSharpHelperMethods.Properties.Settings 3	CSharpHelperMethods.YardimciSiniflar.MetinIslemleri.BoslukKaldir 14
CSharpHelperMethods.Properties.Settings.Default 4	CSharpHelperMethods.YardimciSiniflar.MetinIslemleri.HTMLDuzelt 14
CSharpHelperMethods.Properties.Settings.defaultInstance 4	CSharpHelperMethods.YardimciSiniflar.MetinIslemleri.IdealMetinGetir 15
CSharpHelperMethods.sln 64	CSharpHelperMethods.YardimciSiniflar.MetinIslemleri.MetinIlkKarakterBuyukDigerleriKucukYap 16
CSharpHelperMethods.YardimciSiniflar 4	CSharpHelperMethods.YardimciSiniflar.MetinIslemleri.MetinIlkKarakterleriBuyukYap 16
CSharpHelperMethods.YardimciSiniflar namespace 4	CSharpHelperMethods.YardimciSiniflar.MetinIslemleri.MetinSifrele 16
Classes 4	CSharpHelperMethods.YardimciSiniflar.MetinIslemleri.MetinZararliKarakterDuzelt 17
CSharpHelperMethods.YardimciSiniflar.DataTableIslemleri 5	CSharpHelperMethods.YardimciSiniflar.MetinIslemleri.TelefonNoDuzenle 17
CSharpHelperMethods.YardimciSiniflar.DataTableIslemleri.BaseType 5	CSharpHelperMethods.YardimciSiniflar.MetinIslemleri.text.First 17
CSharpHelperMethods.YardimciSiniflar.DataTableIslemleri.Nullable.GetUnderlyingType 5	CSharpHelperMethods.YardimciSiniflar.MetinIslemleri.text.Substring 17
CSharpHelperMethods.YardimciSiniflar.DataTableIslemleri.OfType.GetGenericTypeDefintion 6	CSharpHelperMethods.YardimciSiniflar.MetinIslemleri.ToString 18
CSharpHelperMethods.YardimciSiniflar.DataTableIslemleri.ToTable 6	CSharpHelperMethods.YardimciSiniflar.MetinIslemleri.ToUpper 18
CSharpHelperMethods.YardimciSiniflar.DataTableIslemleri.typeof 7	CSharpHelperMethods.YardimciSiniflar.MetinIslemleri.TurkceKarakterleriDuzelt 18
CSharpHelperMethods.YardimciSiniflar.DosyaIslemleri 7	CSharpHelperMethods.YardimciSiniflar.MetinIslemleri.UTF8D 18
CSharpHelperMethods.YardimciSiniflar.DosyaIslemleri.DosyaKilitliMi 8	
CSharpHelperMethods.YardimciSiniflar.DosyaIslemleri.DosyaUzantisiUygunMu 9	
CSharpHelperMethods.YardimciSiniflar.DosyaIslemleri.GetirDosyaUzantisiMimeType 9	
CSharpHelperMethods.YardimciSiniflar.DosyaIslemleri.GetirLerikTipiDosyaUzantisille 10	
CSharpHelperMethods.YardimciSiniflar.DosyaIslemleri.GuvenliDizinOlustur	

uzelt 18	CSharpHelperMethods.YardimciSiniflar.TarihIslemleri.Math.Abs 29
CSharpHelperMethods.YardimciSiniflar.SayIslemleri 19	CSharpHelperMethods.YardimciSiniflar.TarihIslemleri.TarihAr aligiHesapla 29
CSharpHelperMethods.YardimciSiniflar.SayIslemleri.AltiHaneliOnayKoduOlustur 19	CSharpHelperMethods.YardimciSiniflar.TarihIslemleri.YasHes apla 30
CSharpHelperMethods.YardimciSiniflar.SayIslemleri.SayiHali neCevir 20	CSharpHelperMethods.YardimciSiniflar.TarihIslemleri.YasHes aplaMetinsel 30
CSharpHelperMethods.YardimciSiniflar.SayIslemleri.SayisalMi 20	CSharpHelperMethods.YardimciSiniflar.TarihIslemleri.Zamani Degistir 31
CSharpHelperMethods.YardimciSiniflar.SayIslemleri.SayiyiYa ziyaCevir 21	
CSharpHelperMethods.YardimciSiniflar.SayIslemleri.text.All 22	<b>D</b>
CSharpHelperMethods.YardimciSiniflar.Sifrelemelslemleri 22	DataTableIslemleri class 5
CSharpHelperMethods.YardimciSiniflar.Sifrelemelslemleri.Hes aplaMD5 22	about DataTableIslemleri class 5
CSharpHelperMethods.YardimciSiniflar.Sifrelemelslemleri.Hes aplaSHA256 23	BaseType 5
CSharpHelperMethods.YardimciSiniflar.Sifrelemelslemleri.Hes aplaSHA512 23	Nullable.GetUnderlyingType 5
CSharpHelperMethods.YardimciSiniflar.Sifrelemelslemleri.Hes aplaSHA512 23	oType.GetGenericTypeDefinition 6
CSharpHelperMethods.YardimciSiniflar.Sifrelemelslemleri.Ras tgeleSifreUret 24	ToDataTable 6
CSharpHelperMethods.YardimciSiniflar.Sifrelemelslemleri.Sifr eUygunMu 24	typeof 7
CSharpHelperMethods.YardimciSiniflar.Siralamalslemleri 25	DataTableIslemleri.cs 64
CSharpHelperMethods.YardimciSiniflar.Siralamalslemleri.Chu nkType 25	Dosyalslemleri class 7
CSharpHelperMethods.YardimciSiniflar.Siralamalslemleri.Chu nkType enumeration 25	about Dosyalslemleri class 7
CSharpHelperMethods.YardimciSiniflar.Siralamalslemleri.Co mpare 25	DosyaKilitliMi 8
CSharpHelperMethods.YardimciSiniflar.Siralamalslemleri.InC hunk 27	DosyaUzantisiUygunMu 9
CSharpHelperMethods.YardimciSiniflar.TarihIslemleri 27	GetirDosyaUzantisiMimeTypeelle 9
CSharpHelperMethods.YardimciSiniflar.TarihIslemleri.AyFarki 28	GetirIcerikTipiDosyaUzantisille 10
CSharpHelperMethods.YardimciSiniflar.TarihIslemleri.KidemT azminatiTarihAraligiHesapla 28	GüvenliDizinOlustur 10
	Dosyalslemleri.cs 66
	<b>F</b>
	Files 62
	FrmAnaSayfa class 32
	about FrmAnaSayfa class 32
	btnAyFarki 34
	btnEPostaDogrula 34
	btnIbanDogrula 34
	btnIlkHarfiBuyukYap 35
	btnIlkHarfleriBuyukYap 35
	btnMd5Hash 35



btnMetinSifrele 35  
 btnMetneDonustur 35  
 btnOnayKoduOlustur 35  
 btnRastgeleSifreUret 35  
 btnSayisalMi 36  
 btnSha256Hash 36  
 btnSha512Hash 36  
 btnSifreUygunMu 36  
 btnTarihAraligi 36  
 btnTcKimlikDogrula 36  
 btnTelefonNoDuzenle 37  
 btnTurkceKarakterDuzelt 37  
 btnYasHesapla 37  
 btnYasHesaplaMetinsel 37  
 cmbIslemTuru 37  
 cmbIslemTuru\_SelectedIndexChanged 42  
 components 37  
 Dispose 51  
 dtpBaslangicTarihi 37  
 dtpBitisTarihi 38  
 FrmAnaSayfa 34  
 FrmAnaSayfa\_Load 51  
 InitializeComponent 52  
 label1 38  
 label10 38  
 label11 38  
 label12 38  
 label13 38  
 label14 38  
 label15 39  
 label16 39  
 label17 39  
 label2 39  
 label3 39  
 label4 39  
 label5 40  
 label6 40  
 label7 40  
 label8 40  
 label9 40  
 lbKisiSonuc 40

lbMetinSonuc 40  
 lbSayiSonuc 41  
 lbSifreSonuc 41  
 lbSonuc 41  
 pnlKisilslemleri 41  
 pnlMetinIslemleri 41  
 pnlSayiIslemleri 41  
 pnlSifreIslemleri 42  
 pnlTarihIslemleri 42  
 txtKisiMetni 42  
 txtMetin 42  
 txtSayi 42  
 txtSifrelenecekMetin 42  
 FrmAnaSayfa.cs 68  
 FrmAnaSayfa.Designer.cs 78

## K

Kisilslemleri class 11  
   about Kisilslemleri class 11  
   DogrulaEPosta 11  
   DogrulaIBAN 12  
   DogrulaTcKimlikNo 12  
 Kisilslemleri.cs 89

## M

MetinIslemleri class 13  
   about MetinIslemleri class 13  
   BoslukKaldir 14  
   HTMLDuzelt 14  
   IdealMetinGetir 15  
   MetinIlkKarakterBuyukDigerleriKucukYap 16  
   MetinIlkKarakterleriBuyukYap 16  
   MetinSifrele 16  
   MetinZararliKarakterDuzelt 17  
   TelefonNoDuzenle 17  
   text.First 17  
   text.Substring 17  
   ToString 18  
   ToUpper 18  
   TurkceKarakterleriDuzelt 18



UTF8Duzelt 18  
MetinIslemleri.cs 91

## N

Numeric enumeration member 25

## P

Program class 62  
    about Program class 62  
    Main 62  
Program.cs 94

## R

Resources class 1  
    about Resources class 1  
    Culture 3  
    resourceCulture 2  
    resourceMan 3  
    ResourceManager 3  
    Resources 2  
Resources.Designer.cs 94

## S

SayIslemleri class 19  
    about SayIslemleri class 19  
    AltHaneliOnayKoduOlustur 19  
    SayiHalineCevir 20  
    SayisalMi 20  
    SayiyiYaziyaCevir 21  
    text.All 22  
SayIslemleri.cs 96  
Settings class 3  
    about Settings class 3  
    Default 4  
    defaultInstance 4  
Settings.Designer.cs 98  
SifrelemIslemleri class 22  
    about SifrelemIslemleri class 22  
    HesaplaMD5 22  
    HesaplaSHA256 23

HesaplaSHA512 23  
RastgeleSifreUret 24  
SifreUygunMu 24  
SifrelemIslemleri.cs 99  
SiralamIslemleri class 25  
    about SiralamIslemleri class 25  
    Compare 25  
    InChunk 27  
SiralamIslemleri.cs 101

## T

TarihIslemleri class 27  
    about TarihIslemleri class 27  
    AyFarki 28  
    KidemTazminatiTarihAraligiHesapla 28  
    Math.Abs 29  
    TarihAraligiHesapla 29  
    YasHesapla 30  
    YasHesaplaMetinsel 30  
    ZamaniDegistir 31  
TarihIslemleri.cs 102