

C Sharp Yardimci Metotlar

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

Table of Contents

Symbol Reference	1
CSharpHelperMethods Namespace	1
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	43
Program Class	61
Program Methods	62
Files	62
AssemblyInfo.cs	63
CSharpHelperMethods.csproj	63
CSharpHelperMethods.sln	63
DataTableIslemleri.cs	64
DosyaIslemleri.cs	66
FrmAnaSayfa.cs	68
FrmAnaSayfa.Designer.cs	77
KisiIslemleri.cs	88
MetinIslemleri.cs	90
Program.cs	93
Resources.Designer.cs	93
SayiIslemleri.cs	95

Settings.Designer.cs	97
Sifrelemelslemleri.cs	98
Siralamalslemleri.cs	100
Tarihlslemleri.cs	102

Index **a**

1 Symbol Reference



1.1 CSharpHelperMethods Namespace

This is namespace CSharpHelperMethods.

Namespaces

Name	Description
Properties (see page 1)	This is namespace CSharpHelperMethods.Properties.
YardimciSiniflar (see page 4)	This is namespace CSharpHelperMethods.YardimciSiniflar.



Classes

	Name	Description
	FrmAnaSayfa (see page 32)	This is class CSharpHelperMethods.FrmAnaSayfa.
	Program (see page 61)	This is class CSharpHelperMethods.Program.

1.1.1 CSharpHelperMethods.Properties Namespace

This is namespace CSharpHelperMethods.Properties.



Classes

	Name	Description
	Resources (see page 1)	A strongly-typed resource class, for looking up localized strings, etc.
	Settings (see page 3)	This is class CSharpHelperMethods.Properties.Settings.

1.1.1.1 Classes

The following table lists classes in this documentation.

Classes

	Name	Description
	Resources (see page 1)	A strongly-typed resource class, for looking up localized strings, etc.
	Settings (see page 3)	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 93)



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

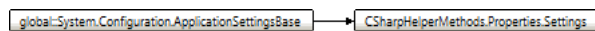
Returns the cached ResourceManager instance used by this class.

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 97](#))

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 (see page 5)	Raporlama için lazım olan DataTable vb. verilerin düzenlenmesini sağlayan sınıflardır.
	DosyaIslemleri (see page 7)	Dosya işlemleri ile ilgili yardımcı sınıf
	KisiIslemleri (see page 11)	Kisi ile ilişkili yardımcı metotların tamamını içeren sınıf
	MetinIslemleri (see page 13)	Metinsel veri kontrol metotlarını barındıran sınıf
	SayiIslemleri (see page 19)	Sayısal veri kontrol metotlarını barındıran sınıf
	Sifrelemelislemleri (see page 22)	Sifre ile ilgili işlemler sınıfı
	Siralamaislemleri (see page 25)	Alfa numerik sıralama için hazırlanmış bir metod
	TarihIslemleri (see page 27)	Tarihsel veri kontrol metotlarını barındıran sınıf

1.1.2.1 Classes

The following table lists classes in this documentation.

Classes

	Name	Description
	DataTablesIslemleri (see page 5)	Raporlama için lazım olan DataTable vb. verilerin düzenlenmesini sağlayan sınıflardır.
	DosyaIslemleri (see page 7)	Dosya işlemleri ile ilgili yardımcı sınıf
	KisiIslemleri (see page 11)	Kisi ile ilişkili yardımcı metotların tamamını içeren sınıf
	MetinIslemleri (see page 13)	Metinsel veri kontrol metotlarını barındıran sınıf
	SayiIslemleri (see page 19)	Sayısal veri kontrol metotlarını barındıran sınıf
	Sifrelemelislemleri (see page 22)	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: }

```

1.1.2.1.1.1.5 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 88)

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 90)

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 yapılmasi 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", "i");
5:     text = text.Replace("ö", "o");
6:     text = text.Replace("ü", "u");
7:     text = text.Replace("s", "s");
8:     text = text.Replace("g", "g");
9:     text = text.Replace("ç", "c");
10:    text = text.Replace("Ü", "U");
11:    text = text.Replace("İ", "I");
12:    text = text.Replace("Ö", "O");
13:    text = text.Replace("Û", "U");
14:    text = text.Replace("S", "S");
15:    text = text.Replace("G", "G");
16:    text = text.Replace("Ç", "C");
17:    return text;
18: }
```

1.1.2.1.4.1.14 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("&ccedil;", "ç");
7:     text = text.Replace("&ouml;", "ö");
8:     text = text.Replace("&uuml;", "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 95)

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:    return sb.ToString();
12: }

```

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, CultureInfo.InvariantCulture, out val))
7:             return 0;
8:         return val / 100;
9:     }
10:
11:     var value = decimal.TryParse(metin.Replace(",", "").Replace(".", "").Replace("_",
""), NumberStyles.Number, CultureInfo.InvariantCulture, out val);
12:     return value ? val : 0;
13: }

```

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 Sayıİslemleri.SayıYazıyaCevir Method

Elimizdeki tutar verilerinin Yazı formatında gösterilmesini sağlayacak bir metottur

C#

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

Parameters

Parameters	Description
decimal tutar	Girilen Sayı Bilgisi

Returns

Sayının Yazıya Dönüştürülmüş Hali

Body Source

```
1: public static string SayıYazıyaCevir(decimal tutar)
2: {
3:     var sTutar = tutar.ToString("F2").Replace('.', ','); // Replace('.', ',') ondalik
    ayracının . olma durumu için
4:     var lira = sTutar.Substring(0, sTutar.IndexOf(',')); //tutarın tam kısmı
5:     var kurus = sTutar.Substring(sTutar.IndexOf(',') + 1, 2);
6:     var yazi = "";
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[] { "KATRİLYON", "TRİLYON", "MİLYAR", "MİLYON", "BİN", "" };
11:
12:    const int grupSayisi = 6;
13:    //sayıdaki 3'lü grup sayısı. katrilyon için 6. (1.234,00 daki grup sayısı 2'dir.)
14:    //KATRİLYON'un basına ekleyeceğiniz her değer için grup sayisini artiriniz.
15:
16:    lira = lira.PadLeft(grupSayisi * 3, '0'); //sayının soluna '0' eklenerek sayı 'grup
    sayısı x 3' basamaklı yapıyor.
17:
18:    for (var i = 0; i < grupSayisi * 3; i += 3) //sayı 3'erli gruplar halinde ele
    aliniyor.
19:    {
20:        var grupDegeri = "";
21:
22:        if (lira.Substring(i, 1) != "0")
23:            grupDegeri += birler[Convert.ToInt32(lira.Substring(i, 1))] + "YÜZ";
    //yüzler
24:
25:        if (grupDegeri == "BİRYÜZ") //biryüz düzeltiliyor.
26:            grupDegeri = "YÜZ";
27:
28:        grupDegeri += onlar[Convert.ToInt32(lira.Substring(i + 1, 1))]; //onlar
29:
30:        grupDegeri += birler[Convert.ToInt32(lira.Substring(i + 2, 1))];
    //birler
31:
32:        if (grupDegeri != "") //binler
33:            grupDegeri += binler[i / 3];
34:
35:        if (grupDegeri == "BİRBİN") //birbin düzeltiliyor.
36:            grupDegeri = "BİN";
37:
38:        yazi += grupDegeri;
39:    }
40:
41:    if (yazi != "")
42:        yazi += " TL ";
43:}
```



```

44:     var yaziUzunlugu = yazi.Length;
45:
46:     if (kurus.Substring(0, 1) != "0") //kurus onlar
47:         yazi += onlar[Convert.ToInt32(kurus.Substring(0, 1))];
48:
49:     if (kurus.Substring(1, 1) != "0") //kurus birler
50:         yazi += birler[Convert.ToInt32(kurus.Substring(1, 1))];
51:
52:     if (yazi.Length > yaziUzunlugu)
53:         yazi += " Kr.";
54:     else
55:         yazi += "SIFIR Kr.";
56:
57:     return yazi;
58: }

```

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 Sifrelemelslemleri Class

Sifre ile ilgili islemler sinifi

Class Hierarchy

CSharpHelperMethods.YardimciSiniflar.Sifrelemelslemleri






C#

```
public static class SifrelemeIslemleri;
```

File

Sifrelemelslemleri.cs (see page 98)

Sifrelemelslemleri Methods

	Name	Description
	HesaplaMD5 (see page 22)	Kullanici sifresinin MD5 yöntemiyle sifrlenmesini saglayan metottur.
	HesaplaSHA256 (see page 23)	Kullanici sifresinin SHA yöntemiyle sifrlenmesini saglayan metottur.
	HesaplaSHA512 (see page 23)	Kullanici sifresinin SHA yöntemiyle sifrlenmesini saglayan metottur.
	RastgeleSifreUret (see page 24)	8 karakter uzunlugunda Rastgele sifre olusturur.
	SifreUygunMu (see page 24)	sifreyi kriterlere göre kontrol edip uymayan kosullarini string listesine geri dönuş yapan methoddur. Uygun olmasi durumunda null döndürecektir.

1.1.2.1.6.1 Sifrelemelslemleri Methods

1.1.2.1.6.1.1 Sifrelemelslemleri.HesaplaMD5 Method

Kullanici sifresinin MD5 yöntemiyle sifrlenmesini saglayan metottur.

C#

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

Parameters

Parameters	Description
string text	Sifrelenecek Metin

Returns

Girdi Metninin Sifrenlenmis 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 sifrenlenmesini saglayan metottur.

C#

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

Parameters

Parameters	Description
string text	Sifrelenecek Metin

Returns

Girdi Metninin Sifrenlenmis Hali

Body Source

```

1: public static string HesaplaSHA256(string text)
2: {
3:     var sha256 = SHA256Managed.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 sifrenlenmesini saglayan metottur.

C#

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

Parameters

Parameters	Description
string text	Sifrelenecek Metin

Returns

Girdi Metninin Sifrelenmis Hali

Body Source

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

1.1.2.1.6.1.4 Sifrelemelslemleri.RastgeleSifreUret Method

8 karakter uzunlugunda Rastgele sifre olusturur.

C#

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

Body Source

```
1: public static string RastgeleSifreUret()
2: {
3:     var sifreUzunluk = 8;
4:     const string gecerliKarakterler = "abcdefghijklmnopqrstuvwxyz1234567890";
5:     var strB = new StringBuilder(100);
6:     var random = new Random();
7:     while (0 < sifreUzunluk--)
8:     {
9:         strB.Append(gecerliKarakterler[random.Next(gecerliKarakterler.Length)]);
10:    }
11:    return strB.ToString();
12: }
```

1.1.2.1.6.1.5 Sifrelemelslemleri.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, @"^(?=.*[!,@,#,$,%,^,&*,?,-,_,~,-,£,(,)])",
RegexOptions.ECMAScript))
16:             mesajlar.Add("Parola en az bir noktalama isareti içermelidir.");
17:
18:         if (!Regex.IsMatch(text, @"^(?=.*[0-9])", RegexOptions.ECMAScript))
19:             mesajlar.Add("Parola en az bir rakam içermelidir.");
20:
21:         if (mesajlar.Count < 1)
22:             return "Uygun";
23:         var sonuc = mesajlar.Aggregate("", (current, item) => current + item + ", ");
24:         sonuc = sonuc.Substring(0, sonuc.Length - 2);
25:         return sonuc;
26: }

```

1.1.2.1.7 Siralamalslemleri Class

Alfa numerik siralama için hazırlanmış bir metod

Class Hierarchy

CSharpHelperMethods.YardimciSiniflar.Siralamalslemleri

C#

```
public static class SiralamaIslemleri;
```



File

Siralamalslemleri.cs (see page 100)

Siralamalslemleri Enumerations

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

Siralamalslemleri Methods

	Name	Description
	Compare (see page 26)	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 100)

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 (string.IsNullOrEmpty(x) || string.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 Siralamalslemleri.InChunk Method

C#

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

Description

This is InChunk, a member of class Siralamalslemleri.

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 metotlarini barindiran sinif

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 arasinda kaç ay oldugunu bulan metottur

	KidemTazminatiTarihAraligiHesapla (see page 28)	Kidem Tazminati Hesaplama Yapilirken ayda kaç gün olursa olsun tamamında çalışmıssa 30 gün üzerinden hesaplanması gerektiği için İsten ayrılıs ve is 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 TariHIslemleri.
	TarihAraligiHesapla (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 Hazirlanan 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.KidemTazminatiTarihAraligiHesapla Method

Kidem Tazminati Hesaplama Yapilirken ayda kaç gün olursa olsun tamamında çalışmıssa 30 gün üzerinden hesaplanması gerektiği için İsten ayrılıs ve is başlangıç tarihleri arasında kaç gün geçtiğini belirtmek için Hesaplama yapılmasını sağlayan metottur.

C#

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

Parameters

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

Returns

Gün Sayisi Bilgisi

Body Source

```
1: public static int KidemTazminatiTarihAraligiHesapla(DateTime? baslangicTarihi,
2: DateTime? bitisTarihi)
3: {
4:     var artacakMi = true;
5:     var result = 0;
6:     var baslangicMi = false;
7:     var bitisMi = false;
```

```

7:     var aylikGunSayisi = 0;
8:     for (var date = (DateTime)baslangicTarihi; date <= (DateTime)bitisTarihi; date =
date.AddDays(1))
9:     {
10:         if (!artacakMi)
11:             result++;
12:         else
13:             artacakMi = false;
14:
15:         if (date.Day == 1)
16:             baslangicMi = true;
17:         else if (date.Day == DateTime.DaysInMonth(date.Year, date.Month))
18:             bitisMi = true;
19:
20:         if (baslangicMi)
21:             aylikGunSayisi++;
22:         if (baslangicMi && bitisMi && aylikGunSayisi != 30)
23:         {
24:             result = result - aylikGunSayisi + 30;
25:             baslangicMi = false;
26:             bitisMi = false;
27:             aylikGunSayisi = 0;
28:         }
29:         else if (bitisMi)
30:         {
31:             baslangicMi = false;
32:             bitisMi = false;
33:             aylikGunSayisi = 0;
34:         }
35:     }
36:
37:     if (baslangicTarihi?.Day > bitisTarihi?.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:     string result;
20:     result = years.ToString().PadLeft(2, '0');
21:     result += " Yil, ";
22:     result += months.ToString().PadLeft(2, '0');
23:     result += " Ay, ";
24:     result += days.ToString().PadLeft(2, '0');
25:     result += " Gün";
26:     return result;
27: }

```

1.1.2.1.8.1.5 TarihIslemleri.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 TarihIslemleri.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 bitiş tarihi üzerinden sorgulama yapıldığında Eger aynı gün için sorgulama yapılıyorsa Bitiş tarihini o günün 23.59.59.999'una ayarlamak için Hazırlanan 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,
```

```



1: 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

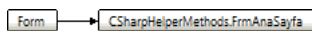
The following table lists classes in this documentation.

Classes

	Name	Description
	FrmAnaSayfa (see page 32)	This is class CSharpHelperMethods.FrmAnaSayfa.
	Program (see page 61)	This is class CSharpHelperMethods.Program.

1.1.3.1 FrmAnaSayfa Class

Class Hierarchy



C#

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


File

FrmAnaSayfa.Designer.cs ([see page 77](#))









Description

This is class CSharpHelperMethods.FrmAnaSayfa.





Methods

	Name	Description
	FrmAnaSayfa (see page 34)	This is FrmAnaSayfa, a member of class FrmAnaSayfa.





FrmAnaSayfa Fields

	Name	Description
	btnAyFarki (see page 34)	This is btnAyFarki, a member of class FrmAnaSayfa.
	btnEPostaDogrula (see page 34)	This is btnEPostaDogrula, a member of class FrmAnaSayfa.
	btnIbanDogrula (see page 35)	This is btnIbanDogrula, a member of class FrmAnaSayfa.
	btnIlkHarfiBuyukYap (see page 35)	This is btnIlkHarfiBuyukYap, a member of class FrmAnaSayfa.
	btnIlkHarfleriBuyukYap (see page 35)	This is btnIlkHarfleriBuyukYap, a member of class FrmAnaSayfa.
	btnMd5Hash (see page 35)	This is btnMd5Hash, a member of class FrmAnaSayfa.
	btnMetinSifrele (see page 35)	This is btnMetinSifrele, a member of class FrmAnaSayfa.
	btnMetneDonustur (see page 35)	This is btnMetneDonustur, a member of class FrmAnaSayfa.

	btnOnayKoduOlustur (see page 35)	This is btnOnayKoduOlustur, a member of class FrmAnaSayfa.
	btnRastgeleSifreUret (see page 36)	This is btnRastgeleSifreUret, a member of class FrmAnaSayfa.
	btnSayisalMi (see page 36)	This is btnSayisalMi, a member of class FrmAnaSayfa.
	btnSha256Hash (see page 36)	This is btnSha256Hash, a member of class FrmAnaSayfa.
	btnSha512Hash (see page 36)	This is btnSha512Hash, a member of class FrmAnaSayfa.
	btnSifreUygunMu (see page 36)	This is btnSifreUygunMu, a member of class FrmAnaSayfa.
	btnTarihAraligi (see page 36)	This is btnTarihAraligi, a member of class FrmAnaSayfa.
	btnTcKimlikDogrula (see page 37)	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 38)	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 39)	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 40)	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 41)	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.
	pnlKisilislemleri (see page 41)	This is pnlKisilislemleri, a member of class FrmAnaSayfa.
	pnlMetinIslemleri (see page 41)	This is pnlMetinIslemleri, a member of class FrmAnaSayfa.
	pnlSayiIslemleri (see page 42)	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
	cmblIslemTuru_SelectedIndexChanged (see page 43)	Islem Türü Combobox'ında seçilen degere göre Ekrandaki panellerden ilgili olan haricinde hepsinin erisilemez hale getirilmesi Eger bir sey seçilmemisse, tamaminin erisilemez hale getirilmesi için Gerekli islemlerin gerçekleştirildiği metottur.
	Dispose (see page 51)	Clean up any resources being used.
	FrmAnaSayfa_Load (see page 51)	Form İlk Yüklendiği zaman ekrandaki bütün panellerin erisilemez hale getirilmesi saglanıyor Bu metotta, çünkü hangi panelin aktifleseceğinin Combobox üzerinden seçilmesi lazım
	InitializeComponent (see page 51)	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 erisilemez 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	
System.EventArgs e	

Body Source

```
1: private void cmbIslemTuru_SelectedIndexChanged(object sender, System.EventArgs e)
2:     {
3:         #region Combobox'tan Seçilen Degere Göre İlgili 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:             // 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: #endregion
70:
71: #region Tarih Islemleri
72:
73:     /// <summary>
74:     /// Paneldeki Baslangiç Tarihi seçili ve tarih girilirse, o tarihten bugüne
kaç yas oldugunun hesaplandigi metottur. Ilgili butona tiklandigi zaman çalisir
75:     /// </summary>
76:     /// <param name="sender"></param>
77:     /// <param name="e"></param>
78:     private void btnYasHesapla_Click(object sender, System.EventArgs e)
79:     {
80:         if (!dtpBaslangicTarihi.Checked)
81:             MessageBox.Show("Baslangiç Tarihi girilmediği için bu işlem
gerçekleştiremez");
82:         else
83:         {
84:             lbSonuc.Items.Clear();
85:             var sb = new StringBuilder();
86:             sb.Append("Dogum tarihi seçilen kisi ");
87:             sb.Append(TarihIslemleri.YasHesapla(dtpBaslangicTarihi.Value));
88:             sb.Append(" yasındadır");
89:             lbSonuc.Items.Add(sb.ToString());
90:         }
91:     }
92:
93:     /// <summary>
94:     /// Paneldeki Baslangiç Tarihi seçili ve tarih girilirse, o tarihten bugüne
kaç yas oldugunun yıl, ay ve gün olarak metinsel bir şekilde hesaplandigi metottur.
95:     /// Ilgili butona tiklandigi zaman çalisir
96:     /// </summary>
97:     /// <param name="sender"></param>
98:     /// <param name="e"></param>
99:     private void btnYasHesaplaMetinsel_Click(object sender, System.EventArgs e)
100:     {
101:         if (!dtpBaslangicTarihi.Checked)
102:             MessageBox.Show("Baslangiç Tarihi girilmediği için bu işlem
gerçekleştiremez");
103:         else
104:         {
105:             lbSonuc.Items.Clear();
106:             var sb = new StringBuilder();
107:             sb.Append(TarihIslemleri.YasHesaplaMetinsel(dtpBaslangicTarihi.Value));

```

```

108:                lbSonuc.Items.Add(sb.ToString());
109:            }
110:        }
111:
112:        /// <summary>
113:        /// Paneldeki Baslangiç Tarihi ve Bitis Tarihi alanlari seçili ve tarih
114:        girilmisse,
115:        /// Söz konusu tarihler arasında kaç yıl, kaç ay ve gün olduğunun bilgisinin
116:        hazirlandigi metottur.
117:        /// Ilgili butona tiklandigi zaman çalışır
118:        /// </summary>
119:        /// <param name="sender"></param>
120:        /// <param name="e"></param>
121:        private void btnTarihAraligi_Click(object sender, System.EventArgs e)
122:        {
123:            if (!dtpBaslangicTarihi.Checked || !dtpBitisTarihi.Checked)
124:                MessageBox.Show("Baslangiç Tarihi veya Bitis Tarihi girilmediği için
125:                bu işlem gerçekleştirilemez");
126:            else if (dtpBaslangicTarihi.Value.Date > dtpBitisTarihi.Value.Date)
127:                MessageBox.Show("Baslangiç Tarihi Bitis Tarihinden büyük olamaz");
128:            else
129:            {
130:                lbSonuc.Items.Clear();
131:                var sb = new StringBuilder();
132:                sb.Append("İki tarih arası: ");
133:                sb.Append(TarihIslemleri.TarihAraligiHesapla(dtpBaslangicTarihi.Value.Date,
134:                dtpBitisTarihi.Value.Date));
135:                lbSonuc.Items.Add(sb.ToString());
136:            }
137:        }
138:
139:        /// <summary>
140:        /// Paneldeki Baslangiç Tarihi ve Bitis Tarihi alanlari seçili ve tarih
141:        girilmisse,
142:        /// Söz konusu tarihler arasında kaç ay olduğunun bilgisinin hazirlandigi
143:        metottur.
144:        /// Ilgili butona tiklandigi zaman çalışır
145:        /// </summary>
146:        /// <param name="sender"></param>
147:        /// <param name="e"></param>
148:        private void btnAyFarki_Click(object sender, System.EventArgs e)
149:        {
150:            if (!dtpBaslangicTarihi.Checked || !dtpBitisTarihi.Checked)
151:                MessageBox.Show("Baslangiç Tarihi veya Bitis Tarihi girilmediği için
152:                bu işlem gerçekleştirilemez");
153:            else if (dtpBaslangicTarihi.Value.Date > dtpBitisTarihi.Value.Date)
154:                MessageBox.Show("Baslangiç Tarihi Bitis Tarihinden büyük olamaz");
155:            else
156:            {
157:                lbSonuc.Items.Clear();
158:                var sb = new StringBuilder();
159:                sb.Append("Ay Farki: ");
160:                sb.Append(TarihIslemleri.AyFarki(dtpBaslangicTarihi.Value.Date,
161:                dtpBitisTarihi.Value.Date));
162:                lbSonuc.Items.Add(sb.ToString());
163:            }
164:        }
165:    #endregion
166:
167:    #region Sifre Islemleri
168:    /// <summary>
169:    /// Paneldeki metin kutusu dolu ise
170:    /// Girilen metnin MD5 algoritması ile sifrelenmis halini
171:    /// Listbox üzerine yazdıran metottur.
172:    /// </summary>
173:    /// <param name="sender"></param>
174:    /// <param name="e"></param>
175:    private void btnMd5Hash_Click(object sender, System.EventArgs e)

```



```

168:         {
169:             if (string.IsNullOrEmpty(txtSifrelenecekMetin.Text.Trim()))
170:                 MessageBox.Show("Sifrelenecek metin girilmediği için bu işlem
gerçekleştirilemez");
171:             else
172:             {
173:                 lbSifreSonuc.Items.Clear();
174:                 var sb = new StringBuilder();
175:                 sb.Append("Girilen Metnin MD5 Karsiligi: ");
176:                 sb.Append(SifrelemeIslemleri.HesaplaMD5(txtSifrelenecekMetin.Text.Trim()));
177:                 lbSifreSonuc.Items.Add(sb.ToString());
178:             }
179:         }
180:
181:         /// <summary>
182:         /// Paneldeki metin kutusu dolu ise
183:         /// Girilen metnin SHA 256 algoritması ile sifrelenmiş halini
184:         /// Listbox üzerine yazdıran metottur.
185:         /// </summary>
186:         /// <param name="sender"></param>
187:         /// <param name="e"></param>
188:         private void btnSha256Hash_Click(object sender, System.EventArgs e)
189:         {
190:             if (string.IsNullOrEmpty(txtSifrelenecekMetin.Text.Trim()))
191:                 MessageBox.Show("Sifrelenecek metin girilmediği için bu işlem
gerçekleştirilemez");
192:             else
193:             {
194:                 lbSifreSonuc.Items.Clear();
195:                 var sb = new StringBuilder();
196:                 sb.Append("Girilen Metnin SHA 256 Karsiligi: ");
197:                 sb.Append(SifrelemeIslemleri.HesaplaSHA256(txtSifrelenecekMetin.Text.Trim()));
198:                 lbSifreSonuc.Items.Add(sb.ToString());
199:             }
200:         }
201:
202:         /// <summary>
203:         /// Paneldeki metin kutusu dolu ise
204:         /// Girilen metnin SHA 512 algoritması ile sifrelenmiş halini
205:         /// Listbox üzerine yazdıran metottur.
206:         /// </summary>
207:         /// <param name="sender"></param>
208:         /// <param name="e"></param>
209:         private void btnSha512Hash_Click(object sender, System.EventArgs e)
210:         {
211:             if (string.IsNullOrEmpty(txtSifrelenecekMetin.Text.Trim()))
212:                 MessageBox.Show("Sifrelenecek metin girilmediği için bu işlem
gerçekleştirilemez");
213:             else
214:             {
215:                 lbSifreSonuc.Items.Clear();
216:                 var sb = new StringBuilder();
217:                 sb.Append("Girilen Metnin SHA 512 Karsiligi: ");
218:                 sb.Append(SifrelemeIslemleri.HesaplaSHA512(txtSifrelenecekMetin.Text.Trim()));
219:                 lbSifreSonuc.Items.Add(sb.ToString());
220:             }
221:         }
222:
223:         /// <summary>
224:         /// Ilgili butona tiklandığında rastgele sifre üretilmesini sağlayan metottur
225:         /// </summary>
226:         /// <param name="sender"></param>
227:         /// <param name="e"></param>
228:         private void btnRastgeleSifreUret_Click(object sender, System.EventArgs e)
229:         {
230:             lbSifreSonuc.Items.Clear();

```

```

231:         var sb = new StringBuilder();
232:         sb.Append("Rastgele Üretilen Sifre: ");
233:         sb.Append(SifrelemeIslemleri.RastgeleSifreUret());
234:         lbSifreSonuc.Items.Add(sb.ToString());
235:     }
236:
237:     /// <summary>
238:     /// Paneldeki metin kutusu dolu ise
239:     /// Burada yazılan metnin bir sifre olarak belirli kriterlere uygun olup
    olmadiginin
240:     /// Gösterilmesini saglayan bir metottur
241:     /// </summary>
242:     /// <param name="sender"></param>
243:     /// <param name="e"></param>
244:     private void btnSifreUygunMu_Click(object sender, System.EventArgs e)
245:     {
246:         if (string.IsNullOrEmpty(txtSifrelenecekMetin.Text.Trim()))
247:             MessageBox.Show("Sifre metni girilmedigi için bu işlem
gerçeklestirilemez");
248:         else
249:         {
250:             lbSifreSonuc.Items.Clear();
251:             var sb = new StringBuilder();
252:             sb.Append("Girilen Sifrenin Uygunluk Durumu: ");
253:             sb.Append(SifrelemeIslemleri.SifreUygunMu(txtSifrelenecekMetin.Text.Trim()));
254:             lbSifreSonuc.Items.Add(sb.ToString());
255:         }
256:     }
257: #endregion
258:
259: #region Sayi Islemleri
260:     /// <summary>
261:     /// Paneldeki metin kutusuna girilen metnin sayisal bir deger olup olmadigini
262:     /// Kontrol eden ve sonucu gösteren metottur.
263:     /// </summary>
264:     /// <param name="sender"></param>
265:     /// <param name="e"></param>
266:     private void btnSayisalMi_Click(object sender, System.EventArgs e)
267:     {
268:         if (string.IsNullOrEmpty(txtSayi.Text.Trim()))
269:             MessageBox.Show("Metin girilmedigi için bu işlem gerçeklestirilemez");
270:         else
271:         {
272:             lbSayiSonuc.Items.Clear();
273:             var sb = new StringBuilder();
274:             sb.Append("Girilen Metin Sayi Mi?: ");
275:             sb.Append(SayiIslemleri.SayisalMi(txtSayi.Text.Trim()) ? "Evet" :
"Hayir");
276:             lbSayiSonuc.Items.Add(sb.ToString());
277:         }
278:     }
279:
280:     /// <summary>
281:     /// Paneldeki metin kutusuna girilen para birimi cinsinden rakamin
282:     /// Yazıya çevrilmesini saglayan ve sonucu listbox üzerinde gösteren bir
metottur.
283:     /// </summary>
284:     /// <param name="sender"></param>
285:     /// <param name="e"></param>
286:     private void btnMetneDonustur_Click(object sender, System.EventArgs e)
287:     {
288:         if (string.IsNullOrEmpty(txtSayi.Text.Trim()))
289:             MessageBox.Show("Metin girilmedigi için bu işlem gerçeklestirilemez");
290:         else if (!SayiIslemleri.SayisalMi(txtSayi.Text.Trim().Replace(".", ""
, ".").Replace(".", "")))
291:             MessageBox.Show("Girilen metin sayisal bir deger olmadigi için bu
işlem gerçeklestirilemez");
292:         else

```

```

293:         {
294:             var metinSadeceSayiKismi = txtSayi.Text.Trim();
295:             var sayi = SayiIslemleri.SayiHalineCevir(metinSadeceSayiKismi);
296:             lbSayiSonuc.Items.Clear();
297:             var sb = new StringBuilder();
298:             sb.Append("Girilen Sayinin Metin Hali (Parasal): ");
299:             sb.Append(SayiIslemleri.SayiyiYaziyaCevir(sayi));
300:             lbSayiSonuc.Items.Add(sb.ToString());
301:         }
302:     }
303:
304:     /// <summary>
305:     /// Rastgele olacak sekilde 6 haneli kod olusturmayi saglayan bir metottur.
306:     /// </summary>
307:     /// <param name="sender"></param>
308:     /// <param name="e"></param>
309:     private void btnOnayKoduOlustur_Click(object sender, System.EventArgs e)
310:     {
311:         lbSayiSonuc.Items.Clear();
312:         var sb = new StringBuilder();
313:         sb.Append("Olusturulan 6 haneli onay kodu: ");
314:         sb.Append(SayiIslemleri.AltiHaneliOnayKoduOlustur());
315:         lbSayiSonuc.Items.Add(sb.ToString());
316:     }
317: #endregion
318:
319: #region Kisi Islemleri
320:     /// <summary>
321:     /// Paneldeki metin kutusuna girilen TC Kimlik numarasinin
322:     /// Standart TC Kimlik No algoritmasina uygun olup olmadigini kontrol eden bir
323:     /// metottur.
324:     /// </summary>
325:     /// <param name="sender"></param>
326:     /// <param name="e"></param>
327:     private void btnTcKimlikDogrula_Click(object sender, System.EventArgs e)
328:     {
329:         if (string.IsNullOrEmpty(txtKisiMetni.Text.Trim()))
330:             MessageBox.Show("Metin girilmedigi için bu islem gerçekleştirilemez");
331:         else if (!SayiIslemleri.SayisalMi(txtKisiMetni.Text.Trim()))
332:             MessageBox.Show("Girilen metin sayisal bir deger olmadigi için bu
333: islem gerçekleştirilemez");
334:         else if (txtKisiMetni.Text.Trim().Length != 11)
335:             MessageBox.Show("Girilen metin 11 karakter olmadigi için bu islem
336: gerçekleştirilemez");
337:         else
338:         {
339:             lbKisiSonuc.Items.Clear();
340:             var sb = new StringBuilder();
341:             sb.Append("TC Kimlik No Dogru Mu?: ");
342:             sb.Append(KisiIslemleri.DogrulaTcKimlikNo(txtKisiMetni.Text.Trim()) ?
343: "Evet" : "Hayir");
344:             lbKisiSonuc.Items.Add(sb.ToString());
345:         }
346:     }
347:
348:     /// <summary>
349:     /// Paneldeki metin kutusuna girilen IBAN Numarasinin
350:     /// Standart IBAN algoritmasina uygun olup olmadigini kontrol eden bir
351:     /// metottur.
352:     /// </summary>
353:     /// <param name="sender"></param>
354:     /// <param name="e"></param>
355:     private void btnIbanDogrula_Click(object sender, System.EventArgs e)
356:     {
357:         if (string.IsNullOrEmpty(txtKisiMetni.Text.Trim()))
358:             MessageBox.Show("Metin girilmedigi için bu islem gerçekleştirilemez");
359:         else if (txtKisiMetni.Text.Trim().Length != 26)
360:             MessageBox.Show("Girilen metin 26 karakter olmadigi için bu islem
361: gerçekleştirilemez. Karakter sayisi: " + txtKisiMetni.Text.Trim().Length);

```

```

356:         else
357:         {
358:             lbKisiSonuc.Items.Clear();
359:             var sb = new StringBuilder();
360:             sb.Append("IBAN Dogru Mu?: ");
361:             sb.Append(KisiIslemleri.DogrulaIBAN(txtKisiMetni.Text.Trim()) ? "Evet"
: "Hayir");
362:             lbKisiSonuc.Items.Add(sb.ToString());
363:         }
364:     }
365:
366:     /// <summary>
367:     /// Paneldeki metin kutusuna girilen metnin
368:     /// Geçerli bir e-posta adresi olup olmadigini kontrol eden bir metottur.
369:     /// </summary>
370:     /// <param name="sender"></param>
371:     /// <param name="e"></param>
372:     private void btnEPostaDogrula_Click(object sender, System.EventArgs e)
373:     {
374:         if (string.IsNullOrEmpty(txtKisiMetni.Text.Trim()))
375:             MessageBox.Show("Metin girilmedigi için bu islem gerçekleştirilemez");
376:         else
377:         {
378:             lbKisiSonuc.Items.Clear();
379:             var sb = new StringBuilder();
380:             sb.Append("E-Posta Dogru Mu?: ");
381:             sb.Append(KisiIslemleri.DogrulaEPosta(txtKisiMetni.Text.Trim()) ?
"Evet" : "Hayir");
382:             lbKisiSonuc.Items.Add(sb.ToString());
383:         }
384:     }
385: #endregion
386:
387: #region Metin Islemleri
388:     /// <summary>
389:     /// Paneldeki metin kutusuna girilen metnin
390:     /// Kelimelerinin ilk harflerini büyük yapmaya yarayan bir metottur.
391:     /// </summary>
392:     /// <param name="sender"></param>
393:     /// <param name="e"></param>
394:     private void btnIlkHarfleriBuyukYap_Click(object sender, System.EventArgs e)
395:     {
396:         if (string.IsNullOrEmpty(txtMetin.Text.Trim()))
397:             MessageBox.Show("Metin girilmedigi için bu islem gerçekleştirilemez");
398:         else
399:         {
400:             lbMetinSonuc.Items.Clear();
401:             var sb = new StringBuilder();
402:             sb.Append("Metnin Düzenlenmis Hali: ");
403:             sb.Append(txtMetin.Text.Trim().MetinIlkKarakterleriBuyukYap());
404:             lbMetinSonuc.Items.Add(sb.ToString());
405:         }
406:     }
407:
408:     /// <summary>
409:     /// Paneldeki metin kutusuna girilen metnin
410:     /// Sadece ilk harfini büyük yapmaya yarayan bir metottur.
411:     /// </summary>
412:     /// <param name="sender"></param>
413:     /// <param name="e"></param>
414:     private void btnIlkHarfiBuyukYap_Click(object sender, System.EventArgs e)
415:     {
416:         if (string.IsNullOrEmpty(txtMetin.Text.Trim()))
417:             MessageBox.Show("Metin girilmedigi için bu islem gerçekleştirilemez");
418:         else
419:         {
420:             lbMetinSonuc.Items.Clear();
421:             var sb = new StringBuilder();
422:             sb.Append("Metnin Düzenlenmis Hali: ");

```

```

423:
sb.Append(MetinIslemleri.MetinIlkKarakterBuyukDigerleriKucukYap(txtMetin.Text.Trim()));
424:         lbMetinSonuc.Items.Add(sb.ToString());
425:     }
426: }
427:
428:     /// <summary>
429:     /// Paneldeki metin kutusuna girilen metnin
430:     /// İçerisindeki Türkçe karakterleri İngilizce hallerine çeviren bir metottur.
431:     /// </summary>
432:     /// <param name="sender"></param>
433:     /// <param name="e"></param>
434:     private void btnTurkceKarakterDuzelt_Click(object sender, System.EventArgs e)
435:     {
436:         if (string.IsNullOrEmpty(txtMetin.Text.Trim()))
437:             MessageBox.Show("Metin girilmediği için bu işlem gerçekleştirilemez");
438:         else
439:         {
440:             lbMetinSonuc.Items.Clear();
441:             var sb = new StringBuilder();
442:             sb.Append("Metnin Düzenlenmiş Hali: ");
443:
sb.Append(MetinIslemleri.TurkceKarakterleriDuzelt(txtMetin.Text.Trim()));
444:             lbMetinSonuc.Items.Add(sb.ToString());
445:         }
446:     }
447:
448:     /// <summary>
449:     /// KVKK durumlarından dolayı TC Kimlik No ve Vergi No gibi alanların
450:     /// Matbu çıktılarda gizlenmesi gibi bir durum söz konusu olduğundan
451:     /// Girilen bu 10 veya 11 karakterlik metinleri şifrelemeye yarayan bir
metottur
452:     /// </summary>
453:     /// <param name="sender"></param>
454:     /// <param name="e"></param>
455:     private void btnMetinSifrele_Click(object sender, System.EventArgs e)
456:     {
457:         if (string.IsNullOrEmpty(txtMetin.Text.Trim()))
458:             MessageBox.Show("Metin girilmediği için bu işlem gerçekleştirilemez");
459:         else if (!SayiIslemleri.SayisalMi(txtMetin.Text.Trim()))
460:             MessageBox.Show("Girilen metin sayısal bir değer olmadığı için bu
işlem gerçekleştirilemez");
461:         else if (txtMetin.Text.Trim().Length != 11 && txtMetin.Text.Trim().Length
!= 10)
462:             MessageBox.Show("Girilen metin 10 veya 11 karakter olmadığı için bu
işlem gerçekleştirilemez");
463:         else
464:         {
465:             lbMetinSonuc.Items.Clear();
466:             var sb = new StringBuilder();
467:             sb.Append("TC Kimlik No veya Vergi No'nun Sifrelenmiş Hali: ");
468:             sb.Append(MetinIslemleri.MetinSifrele(txtMetin.Text.Trim()));
469:             lbMetinSonuc.Items.Add(sb.ToString());
470:         }
471:     }
472:
473:     /// <summary>
474:     /// Ekrandan çeşitli karakterler içeren telefon numaraları alındığı zaman
475:     /// Bu telefon numarasını olması gerektiği hale çevirmeye yarayan bir metottur.
476:     /// </summary>
477:     /// <param name="sender"></param>
478:     /// <param name="e"></param>
479:     private void btnTelefonNoDuzenle_Click(object sender, System.EventArgs e)
480:     {
481:         if (string.IsNullOrEmpty(txtMetin.Text.Trim()))
482:             MessageBox.Show("Metin girilmediği için bu işlem gerçekleştirilemez");
483:         else
484:         {
485:             lbMetinSonuc.Items.Clear();

```

```

486:                var sb = new StringBuilder();
487:                sb.Append("Telefon Numarasinin Düzenlenmis Hali: ");
488:                sb.Append(MetinIslemleri.TelefonNoDuzenle(txtMetin.Text.Trim()));
489:                lbMetinSonuc.Items.Add(sb.ToString());
490:            }
491:        }
492:    #endregion
493: }
494: }

```

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 aktifleşeceğinin Combobox üzerinden seçilmesi lazım

C#

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

Parameters

Parameters	Description
object sender	
System.EventArgs e	

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: }

```

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.txtSifrelenecekMetin = new System.Windows.Forms.TextBox();
27:    this.label5 = new System.Windows.Forms.Label();
28:    this.pnlSayiIslemleri = new System.Windows.Forms.Panel();
29:    this.label9 = new System.Windows.Forms.Label();
30:    this.btnOnayKoduOlustur = new System.Windows.Forms.Button();
31:    this.btnMetneDonustur = new System.Windows.Forms.Button();
32:    this.label10 = new System.Windows.Forms.Label();
33:    this.lbSayiSonuc = new System.Windows.Forms.ListBox();
34:    this.btnSayisalMi = new System.Windows.Forms.Button();
35:    this.txtSayi = new System.Windows.Forms.TextBox();
36:    this.label11 = new System.Windows.Forms.Label();
37:    this.pnlKisiIslemleri = new System.Windows.Forms.Panel();
38:    this.label12 = new System.Windows.Forms.Label();
39:    this.btnEPostaDogrula = new System.Windows.Forms.Button();
40:    this.btnIbanDogrula = new System.Windows.Forms.Button();
41:    this.label13 = new System.Windows.Forms.Label();
42:    this.lbKisiSonuc = new System.Windows.Forms.ListBox();
43:    this.btnTcKimlikDogrula = new System.Windows.Forms.Button();
44:    this.txtKisiMetni = new System.Windows.Forms.TextBox();
45:    this.label14 = new System.Windows.Forms.Label();
46:    this.pnlMetinIslemleri = new System.Windows.Forms.Panel();
47:    this.btnTelefonNoDuzenle = new System.Windows.Forms.Button();
48:    this.btnMetinSifrele = new System.Windows.Forms.Button();
49:    this.label15 = new System.Windows.Forms.Label();
50:    this.btnTurkceKarakterDuzelt = new System.Windows.Forms.Button();
51:    this.btnIlkHarfiBuyukYap = new System.Windows.Forms.Button();
52:    this.label16 = new System.Windows.Forms.Label();
53:    this.lbMetinSonuc = new System.Windows.Forms.ListBox();
54:    this.btnIlkHarfleriBuyukYap = new System.Windows.Forms.Button();
55:    this.txtMetin = new System.Windows.Forms.TextBox();
56:    this.label17 = new System.Windows.Forms.Label();
57:    this.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[] {
68:             "Metin Islemleri",
69:             "Tarih Islemleri",
70:             "Sayi Islemleri",
71:             "Sifre Islemleri",
72:             "Kisi Islemleri"});
73:         this.cmbIslemTuru.Location = new System.Drawing.Point(132, 13);
74:         this.cmbIslemTuru.Name = "cmbIslemTuru";
75:         this.cmbIslemTuru.Size = new System.Drawing.Size(200, 21);
76:         this.cmbIslemTuru.TabIndex = 0;
77:         this.cmbIslemTuru.SelectedIndexChanged += new
System.EventHandler(this.cmbIslemTuru_SelectedIndexChanged);
78:         //
79:         // label1
80:         //
81:         this.label1.AutoSize = true;
82:         this.label1.Location = new System.Drawing.Point(12, 16);
83:         this.label1.Name = "label1";
84:         this.label1.Size = new System.Drawing.Size(93, 13);
85:         this.label1.TabIndex = 1;
86:         this.label1.Text = "Islem Türü Seçiniz";
87:         //
88:         // pnlTarihIslemleri
89:         //
90:         this.pnlTarihIslemleri.BorderStyle = System.Windows.Forms.BorderStyle.FixedSingle;
91:         this.pnlTarihIslemleri.Controls.Add(this.btnAyFarki);
92:         this.pnlTarihIslemleri.Controls.Add(this.label8);
93:         this.pnlTarihIslemleri.Controls.Add(this.btnTarihAraligi);
94:         this.pnlTarihIslemleri.Controls.Add(this.btnYasHesaplaMetinsel);
95:         this.pnlTarihIslemleri.Controls.Add(this.btnYasHesapla);
96:         this.pnlTarihIslemleri.Controls.Add(this.label4);
97:         this.pnlTarihIslemleri.Controls.Add(this.lbSonuc);
98:         this.pnlTarihIslemleri.Controls.Add(this.dtpBitisTarihi);
99:         this.pnlTarihIslemleri.Controls.Add(this.dtpBaslangicTarihi);
100:        this.pnlTarihIslemleri.Controls.Add(this.label3);
101:        this.pnlTarihIslemleri.Controls.Add(this.label2);
102:        this.pnlTarihIslemleri.Location = new System.Drawing.Point(15, 56);
103:        this.pnlTarihIslemleri.Name = "pnlTarihIslemleri";
104:        this.pnlTarihIslemleri.Size = new System.Drawing.Size(374, 181);
105:        this.pnlTarihIslemleri.TabIndex = 2;
106:        //
107:        // btnAyFarki
108:        //
109:        this.btnAyFarki.Location = new System.Drawing.Point(286, 76);
110:        this.btnAyFarki.Name = "btnAyFarki";
111:        this.btnAyFarki.Size = new System.Drawing.Size(79, 23);
112:        this.btnAyFarki.TabIndex = 12;
113:        this.btnAyFarki.Text = "Ay Farki";
114:        this.btnAyFarki.UseVisualStyleBackColor = true;
115:        this.btnAyFarki.Click += new System.EventHandler(this.btnAyFarki_Click);
116:        //
117:        // label8
118:        //
119:        this.label8.AutoSize = true;
120:        this.label8.Dock = System.Windows.Forms.DockStyle.Top;
121:        this.label8.Location = new System.Drawing.Point(0, 0);
122:        this.label8.Name = "label8";
123:        this.label8.Size = new System.Drawing.Size(71, 13);
124:        this.label8.TabIndex = 11;
125:        this.label8.Text = "Tarih Islemleri";
126:        //
127:        // btnTarihAraligi
128:        //
129:        this.btnTarihAraligi.Location = new System.Drawing.Point(204, 76);
130:        this.btnTarihAraligi.Name = "btnTarihAraligi";
131:        this.btnTarihAraligi.Size = new System.Drawing.Size(76, 23);

```



```
132:         this.btnTarihAraligi.TabIndex = 10;
133:         this.btnTarihAraligi.Text = "Tarih Araligi";
134:         this.btnTarihAraligi.UseVisualStyleBackColor = true;
135:         this.btnTarihAraligi.Click += new System.EventHandler(this.btnTarihAraligi_Click);
136:         //
137:         // btnYasHesaplaMetinsel
138:         //
139:         this.btnYasHesaplaMetinsel.Location = new System.Drawing.Point(84, 76);
140:         this.btnYasHesaplaMetinsel.Name = "btnYasHesaplaMetinsel";
141:         this.btnYasHesaplaMetinsel.Size = new System.Drawing.Size(113, 23);
142:         this.btnYasHesaplaMetinsel.TabIndex = 9;
143:         this.btnYasHesaplaMetinsel.Text = "Yas Hesapla (Metin)";
144:         this.btnYasHesaplaMetinsel.UseVisualStyleBackColor = true;
145:         this.btnYasHesaplaMetinsel.Click += new
System.EventHandler(this.btnYasHesaplaMetinsel_Click);
146:         //
147:         // btnYasHesapla
148:         //
149:         this.btnYasHesapla.Location = new System.Drawing.Point(3, 76);
150:         this.btnYasHesapla.Name = "btnYasHesapla";
151:         this.btnYasHesapla.Size = new System.Drawing.Size(75, 23);
152:         this.btnYasHesapla.TabIndex = 8;
153:         this.btnYasHesapla.Text = "Yas Hesapla";
154:         this.btnYasHesapla.UseVisualStyleBackColor = true;
155:         this.btnYasHesapla.Click += new System.EventHandler(this.btnYasHesapla_Click);
156:         //
157:         // label4
158:         //
159:         this.label4.AutoSize = true;
160:         this.label4.Location = new System.Drawing.Point(51, 109);
161:         this.label4.Name = "label4";
162:         this.label4.Size = new System.Drawing.Size(38, 13);
163:         this.label4.TabIndex = 7;
164:         this.label4.Text = "Sonuç";
165:         //
166:         // lbSonuc
167:         //
168:         this.lbSonuc.FormattingEnabled = true;
169:         this.lbSonuc.HorizontalScrollbar = true;
170:         this.lbSonuc.Location = new System.Drawing.Point(117, 109);
171:         this.lbSonuc.Name = "lbSonuc";
172:         this.lbSonuc.Size = new System.Drawing.Size(200, 69);
173:         this.lbSonuc.TabIndex = 6;
174:         //
175:         // dtpBitisTarihi
176:         //
177:         this.dtpBitisTarihi.Format = System.Windows.Forms.DateTimePickerFormat.Short;
178:         this.dtpBitisTarihi.Location = new System.Drawing.Point(117, 43);
179:         this.dtpBitisTarihi.Name = "dtpBitisTarihi";
180:         this.dtpBitisTarihi.ShowCheckBox = true;
181:         this.dtpBitisTarihi.Size = new System.Drawing.Size(200, 20);
182:         this.dtpBitisTarihi.TabIndex = 5;
183:         //
184:         // dtpBaslangicTarihi
185:         //
186:         this.dtpBaslangicTarihi.Format = System.Windows.Forms.DateTimePickerFormat.Short;
187:         this.dtpBaslangicTarihi.Location = new System.Drawing.Point(117, 17);
188:         this.dtpBaslangicTarihi.Name = "dtpBaslangicTarihi";
189:         this.dtpBaslangicTarihi.ShowCheckBox = true;
190:         this.dtpBaslangicTarihi.Size = new System.Drawing.Size(200, 20);
191:         this.dtpBaslangicTarihi.TabIndex = 4;
192:         //
193:         // label3
194:         //
195:         this.label3.AutoSize = true;
196:         this.label3.Location = new System.Drawing.Point(44, 47);
197:         this.label3.Name = "label3";
198:         this.label3.Size = new System.Drawing.Size(55, 13);
199:         this.label3.TabIndex = 3;
```

```
200:         this.label3.Text = "Bitis Tarihi";
201:         //
202:         // label2
203:         //
204:         this.label2.AutoSize = true;
205:         this.label2.Location = new System.Drawing.Point(17, 24);
206:         this.label2.Name = "label2";
207:         this.label2.Size = new System.Drawing.Size(82, 13);
208:         this.label2.TabIndex = 2;
209:         this.label2.Text = "Baslangiç Tarihi";
210:         //
211:         // pnlSifreIslemleri
212:         //
213:         this.pnlSifreIslemleri.BorderStyle = System.Windows.Forms.BorderStyle.FixedSingle;
214:         this.pnlSifreIslemleri.Controls.Add(this.label7);
215:         this.pnlSifreIslemleri.Controls.Add(this.btnSifreUygunMu);
216:         this.pnlSifreIslemleri.Controls.Add(this.btnRastgeleSifreUret);
217:         this.pnlSifreIslemleri.Controls.Add(this.btnSha512Hash);
218:         this.pnlSifreIslemleri.Controls.Add(this.btnSha256Hash);
219:         this.pnlSifreIslemleri.Controls.Add(this.label6);
220:         this.pnlSifreIslemleri.Controls.Add(this.lbSifreSonuc);
221:         this.pnlSifreIslemleri.Controls.Add(this.btnMd5Hash);
222:         this.pnlSifreIslemleri.Controls.Add(this.txtSifrelenecekMetin);
223:         this.pnlSifreIslemleri.Controls.Add(this.label5);
224:         this.pnlSifreIslemleri.Location = new System.Drawing.Point(411, 56);
225:         this.pnlSifreIslemleri.Name = "pnlSifreIslemleri";
226:         this.pnlSifreIslemleri.Size = new System.Drawing.Size(377, 181);
227:         this.pnlSifreIslemleri.TabIndex = 3;
228:         //
229:         // label7
230:         //
231:         this.label7.AutoSize = true;
232:         this.label7.Dock = System.Windows.Forms.DockStyle.Top;
233:         this.label7.Location = new System.Drawing.Point(0, 0);
234:         this.label7.Name = "label7";
235:         this.label7.Size = new System.Drawing.Size(68, 13);
236:         this.label7.TabIndex = 18;
237:         this.label7.Text = "Sifre Islemleri";
238:         this.label7.TextAlign = System.Drawing.ContentAlignment.TopCenter;
239:         //
240:         // btnSifreUygunMu
241:         //
242:         this.btnSifreUygunMu.Location = new System.Drawing.Point(173, 71);
243:         this.btnSifreUygunMu.Name = "btnSifreUygunMu";
244:         this.btnSifreUygunMu.Size = new System.Drawing.Size(145, 23);
245:         this.btnSifreUygunMu.TabIndex = 17;
246:         this.btnSifreUygunMu.Text = "Sifre Uygun Mu?";
247:         this.btnSifreUygunMu.UseVisualStyleBackColor = true;
248:         this.btnSifreUygunMu.Click += new System.EventHandler(this.btnSifreUygunMu_Click);
249:         //
250:         // btnRastgeleSifreUret
251:         //
252:         this.btnRastgeleSifreUret.Location = new System.Drawing.Point(21, 71);
253:         this.btnRastgeleSifreUret.Name = "btnRastgeleSifreUret";
254:         this.btnRastgeleSifreUret.Size = new System.Drawing.Size(145, 23);
255:         this.btnRastgeleSifreUret.TabIndex = 16;
256:         this.btnRastgeleSifreUret.Text = "Rastgele Sifre Üret";
257:         this.btnRastgeleSifreUret.UseVisualStyleBackColor = true;
258:         this.btnRastgeleSifreUret.Click += new
System.EventHandler(this.btnRastgeleSifreUret_Click);
259:         //
260:         // btnSha512Hash
261:         //
262:         this.btnSha512Hash.Location = new System.Drawing.Point(243, 42);
263:         this.btnSha512Hash.Name = "btnSha512Hash";
264:         this.btnSha512Hash.Size = new System.Drawing.Size(75, 23);
265:         this.btnSha512Hash.TabIndex = 15;
266:         this.btnSha512Hash.Text = "SHA 512";
267:         this.btnSha512Hash.UseVisualStyleBackColor = true;
```

```
268:         this.btnSha512Hash.Click += new System.EventHandler(this.btnSha512Hash_Click);
269:         //
270:         // btnSha256Hash
271:         //
272:         this.btnSha256Hash.Location = new System.Drawing.Point(131, 42);
273:         this.btnSha256Hash.Name = "btnSha256Hash";
274:         this.btnSha256Hash.Size = new System.Drawing.Size(75, 23);
275:         this.btnSha256Hash.TabIndex = 14;
276:         this.btnSha256Hash.Text = "SHA 256";
277:         this.btnSha256Hash.UseVisualStyleBackColor = true;
278:         this.btnSha256Hash.Click += new System.EventHandler(this.btnSha256Hash_Click);
279:         //
280:         // label6
281:         //
282:         this.label6.AutoSize = true;
283:         this.label6.Location = new System.Drawing.Point(74, 104);
284:         this.label6.Name = "label6";
285:         this.label6.Size = new System.Drawing.Size(38, 13);
286:         this.label6.TabIndex = 13;
287:         this.label6.Text = "Sonuç";
288:         //
289:         // lbSifreSonuc
290:         //
291:         this.lbSifreSonuc.FormattingEnabled = true;
292:         this.lbSifreSonuc.HorizontalScrollbar = true;
293:         this.lbSifreSonuc.Location = new System.Drawing.Point(118, 100);
294:         this.lbSifreSonuc.Name = "lbSifreSonuc";
295:         this.lbSifreSonuc.Size = new System.Drawing.Size(200, 69);
296:         this.lbSifreSonuc.TabIndex = 12;
297:         //
298:         // btnMd5Hash
299:         //
300:         this.btnMd5Hash.Location = new System.Drawing.Point(21, 42);
301:         this.btnMd5Hash.Name = "btnMd5Hash";
302:         this.btnMd5Hash.Size = new System.Drawing.Size(75, 23);
303:         this.btnMd5Hash.TabIndex = 11;
304:         this.btnMd5Hash.Text = "MD5";
305:         this.btnMd5Hash.UseVisualStyleBackColor = true;
306:         this.btnMd5Hash.Click += new System.EventHandler(this.btnMd5Hash_Click);
307:         //
308:         // txtSifrelenecekMetin
309:         //
310:         this.txtSifrelenecekMetin.Location = new System.Drawing.Point(118, 16);
311:         this.txtSifrelenecekMetin.Name = "txtSifrelenecekMetin";
312:         this.txtSifrelenecekMetin.Size = new System.Drawing.Size(199, 20);
313:         this.txtSifrelenecekMetin.TabIndex = 4;
314:         //
315:         // label5
316:         //
317:         this.label5.AutoSize = true;
318:         this.label5.Location = new System.Drawing.Point(17, 19);
319:         this.label5.Name = "label5";
320:         this.label5.Size = new System.Drawing.Size(95, 13);
321:         this.label5.TabIndex = 3;
322:         this.label5.Text = "Sifrelenecek Metin";
323:         //
324:         // pnlSayiIslemleri
325:         //
326:         this.pnlSayiIslemleri.BorderStyle = System.Windows.Forms.BorderStyle.FixedSingle;
327:         this.pnlSayiIslemleri.Controls.Add(this.label9);
328:         this.pnlSayiIslemleri.Controls.Add(this.btnOnayKoduOlustur);
329:         this.pnlSayiIslemleri.Controls.Add(this.btnMetneDonustur);
330:         this.pnlSayiIslemleri.Controls.Add(this.label10);
331:         this.pnlSayiIslemleri.Controls.Add(this.lbSayiSonuc);
332:         this.pnlSayiIslemleri.Controls.Add(this.btnSayisalMi);
333:         this.pnlSayiIslemleri.Controls.Add(this.txtSayi);
334:         this.pnlSayiIslemleri.Controls.Add(this.label11);
335:         this.pnlSayiIslemleri.Location = new System.Drawing.Point(15, 257);
336:         this.pnlSayiIslemleri.Name = "pnlSayiIslemleri";
```

```
337:         this.pnlSayiIslemleri.Size = new System.Drawing.Size(377, 181);
338:         this.pnlSayiIslemleri.TabIndex = 19;
339:         //
340:         // label9
341:         //
342:         this.label9.AutoSize = true;
343:         this.label9.Dock = System.Windows.Forms.DockStyle.Top;
344:         this.label9.Location = new System.Drawing.Point(0, 0);
345:         this.label9.Name = "label9";
346:         this.label9.Size = new System.Drawing.Size(67, 13);
347:         this.label9.TabIndex = 18;
348:         this.label9.Text = "Sayi Islemleri";
349:         this.label9.TextAlign = System.Drawing.ContentAlignment.TopCenter;
350:         //
351:         // btnOnayKoduOlustur
352:         //
353:         this.btnOnayKoduOlustur.Location = new System.Drawing.Point(243, 42);
354:         this.btnOnayKoduOlustur.Name = "btnOnayKoduOlustur";
355:         this.btnOnayKoduOlustur.Size = new System.Drawing.Size(75, 23);
356:         this.btnOnayKoduOlustur.TabIndex = 15;
357:         this.btnOnayKoduOlustur.Text = "Onay Kodu";
358:         this.btnOnayKoduOlustur.UseVisualStyleBackColor = true;
359:         this.btnOnayKoduOlustur.Click += new
System.EventHandler(this.btnOnayKoduOlustur_Click);
360:         //
361:         // btnMetneDonustur
362:         //
363:         this.btnMetneDonustur.Location = new System.Drawing.Point(102, 42);
364:         this.btnMetneDonustur.Name = "btnMetneDonustur";
365:         this.btnMetneDonustur.Size = new System.Drawing.Size(135, 23);
366:         this.btnMetneDonustur.TabIndex = 14;
367:         this.btnMetneDonustur.Text = "Para Metnine Dönüştür";
368:         this.btnMetneDonustur.UseVisualStyleBackColor = true;
369:         this.btnMetneDonustur.Click += new
System.EventHandler(this.btnMetneDonustur_Click);
370:         //
371:         // label10
372:         //
373:         this.label10.AutoSize = true;
374:         this.label10.Location = new System.Drawing.Point(18, 78);
375:         this.label10.Name = "label10";
376:         this.label10.Size = new System.Drawing.Size(38, 13);
377:         this.label10.TabIndex = 13;
378:         this.label10.Text = "Sonuç";
379:         //
380:         // lbSayiSonuc
381:         //
382:         this.lbSayiSonuc.FormattingEnabled = true;
383:         this.lbSayiSonuc.HorizontalScrollbar = true;
384:         this.lbSayiSonuc.Location = new System.Drawing.Point(62, 78);
385:         this.lbSayiSonuc.Name = "lbSayiSonuc";
386:         this.lbSayiSonuc.Size = new System.Drawing.Size(254, 82);
387:         this.lbSayiSonuc.TabIndex = 12;
388:         //
389:         // btnSayisalMi
390:         //
391:         this.btnSayisalMi.Location = new System.Drawing.Point(21, 42);
392:         this.btnSayisalMi.Name = "btnSayisalMi";
393:         this.btnSayisalMi.Size = new System.Drawing.Size(75, 23);
394:         this.btnSayisalMi.TabIndex = 11;
395:         this.btnSayisalMi.Text = "Sayisal Mi?";
396:         this.btnSayisalMi.UseVisualStyleBackColor = true;
397:         this.btnSayisalMi.Click += new System.EventHandler(this.btnSayisalMi_Click);
398:         //
399:         // txtSayi
400:         //
401:         this.txtSayi.Location = new System.Drawing.Point(118, 16);
402:         this.txtSayi.Name = "txtSayi";
403:         this.txtSayi.Size = new System.Drawing.Size(199, 20);
```

```
404:         this.txtSayi.TabIndex = 4;
405:         //
406:         // label11
407:         //
408:         this.label11.AutoSize = true;
409:         this.label11.Location = new System.Drawing.Point(17, 19);
410:         this.label11.Name = "label11";
411:         this.label11.Size = new System.Drawing.Size(95, 13);
412:         this.label11.TabIndex = 3;
413:         this.label11.Text = "Sifrelenecek Metin";
414:         //
415:         // pnlKisiIslemleri
416:         //
417:         this.pnlKisiIslemleri.BorderStyle = System.Windows.Forms.BorderStyle.FixedSingle;
418:         this.pnlKisiIslemleri.Controls.Add(this.label12);
419:         this.pnlKisiIslemleri.Controls.Add(this.btnEPostaDogrula);
420:         this.pnlKisiIslemleri.Controls.Add(this.btnIbanDogrula);
421:         this.pnlKisiIslemleri.Controls.Add(this.label13);
422:         this.pnlKisiIslemleri.Controls.Add(this.lbKisiSonuc);
423:         this.pnlKisiIslemleri.Controls.Add(this.btnTcKimlikDogrula);
424:         this.pnlKisiIslemleri.Controls.Add(this.txtKisiMetni);
425:         this.pnlKisiIslemleri.Controls.Add(this.label14);
426:         this.pnlKisiIslemleri.Location = new System.Drawing.Point(411, 257);
427:         this.pnlKisiIslemleri.Name = "pnlKisiIslemleri";
428:         this.pnlKisiIslemleri.Size = new System.Drawing.Size(377, 181);
429:         this.pnlKisiIslemleri.TabIndex = 20;
430:         //
431:         // label12
432:         //
433:         this.label12.AutoSize = true;
434:         this.label12.Dock = System.Windows.Forms.DockStyle.Top;
435:         this.label12.Location = new System.Drawing.Point(0, 0);
436:         this.label12.Name = "label12";
437:         this.label12.Size = new System.Drawing.Size(63, 13);
438:         this.label12.TabIndex = 18;
439:         this.label12.Text = "Kisi Islemleri";
440:         this.label12.TextAlign = System.Drawing.ContentAlignment.TopCenter;
441:         //
442:         // btnEPostaDogrula
443:         //
444:         this.btnEPostaDogrula.Location = new System.Drawing.Point(223, 42);
445:         this.btnEPostaDogrula.Name = "btnEPostaDogrula";
446:         this.btnEPostaDogrula.Size = new System.Drawing.Size(95, 23);
447:         this.btnEPostaDogrula.TabIndex = 15;
448:         this.btnEPostaDogrula.Text = "E-Posta Dogrula";
449:         this.btnEPostaDogrula.UseVisualStyleBackColor = true;
450:         this.btnEPostaDogrula.Click += new
System.EventHandler(this.btnEPostaDogrula_Click);
451:         //
452:         // btnIbanDogrula
453:         //
454:         this.btnIbanDogrula.Location = new System.Drawing.Point(131, 42);
455:         this.btnIbanDogrula.Name = "btnIbanDogrula";
456:         this.btnIbanDogrula.Size = new System.Drawing.Size(86, 23);
457:         this.btnIbanDogrula.TabIndex = 14;
458:         this.btnIbanDogrula.Text = "IBAN Dogrula";
459:         this.btnIbanDogrula.UseVisualStyleBackColor = true;
460:         this.btnIbanDogrula.Click += new System.EventHandler(this.btnIbanDogrula_Click);
461:         //
462:         // label13
463:         //
464:         this.label13.AutoSize = true;
465:         this.label13.Location = new System.Drawing.Point(18, 78);
466:         this.label13.Name = "label13";
467:         this.label13.Size = new System.Drawing.Size(38, 13);
468:         this.label13.TabIndex = 13;
469:         this.label13.Text = "Sonuç";
470:         //
471:         // lbKisiSonuc
```



```

472:    //
473:    this.lbKisiSonuc.FormattingEnabled = true;
474:    this.lbKisiSonuc.HorizontalScrollbar = true;
475:    this.lbKisiSonuc.Location = new System.Drawing.Point(62, 78);
476:    this.lbKisiSonuc.Name = "lbKisiSonuc";
477:    this.lbKisiSonuc.Size = new System.Drawing.Size(254, 82);
478:    this.lbKisiSonuc.TabIndex = 12;
479:    //
480:    // btnTcKimlikDogrula
481:    //
482:    this.btnTcKimlikDogrula.Location = new System.Drawing.Point(21, 42);
483:    this.btnTcKimlikDogrula.Name = "btnTcKimlikDogrula";
484:    this.btnTcKimlikDogrula.Size = new System.Drawing.Size(104, 23);
485:    this.btnTcKimlikDogrula.TabIndex = 11;
486:    this.btnTcKimlikDogrula.Text = "TC Kimlik Dogrula";
487:    this.btnTcKimlikDogrula.UseVisualStyleBackColor = true;
488:    this.btnTcKimlikDogrula.Click += new
System.EventHandler(this.btnTcKimlikDogrula_Click);
489:    //
490:    // txtKisiMetni
491:    //
492:    this.txtKisiMetni.Location = new System.Drawing.Point(118, 16);
493:    this.txtKisiMetni.Name = "txtKisiMetni";
494:    this.txtKisiMetni.Size = new System.Drawing.Size(199, 20);
495:    this.txtKisiMetni.TabIndex = 4;
496:    //
497:    // label14
498:    //
499:    this.label14.AutoSize = true;
500:    this.label14.Location = new System.Drawing.Point(17, 19);
501:    this.label14.Name = "label14";
502:    this.label14.Size = new System.Drawing.Size(33, 13);
503:    this.label14.TabIndex = 3;
504:    this.label14.Text = "Metin";
505:    //
506:    // pnlMetinIslemleri
507:    //
508:    this.pnlMetinIslemleri.BorderStyle = System.Windows.Forms.BorderStyle.FixedSingle;
509:    this.pnlMetinIslemleri.Controls.Add(this.btnTelefonNoDuzenle);
510:    this.pnlMetinIslemleri.Controls.Add(this.btnMetinSifrele);
511:    this.pnlMetinIslemleri.Controls.Add(this.label15);
512:    this.pnlMetinIslemleri.Controls.Add(this.btnTurkceKarakterDuzelt);
513:    this.pnlMetinIslemleri.Controls.Add(this.btnIlkHarfiBuyukYap);
514:    this.pnlMetinIslemleri.Controls.Add(this.label16);
515:    this.pnlMetinIslemleri.Controls.Add(this.lbMetinSonuc);
516:    this.pnlMetinIslemleri.Controls.Add(this.btnIlkHarfleriBuyukYap);
517:    this.pnlMetinIslemleri.Controls.Add(this.txtMetin);
518:    this.pnlMetinIslemleri.Controls.Add(this.label17);
519:    this.pnlMetinIslemleri.Location = new System.Drawing.Point(15, 458);
520:    this.pnlMetinIslemleri.Name = "pnlMetinIslemleri";
521:    this.pnlMetinIslemleri.Size = new System.Drawing.Size(773, 181);
522:    this.pnlMetinIslemleri.TabIndex = 21;
523:    //
524:    // btnTelefonNoDuzenle
525:    //
526:    this.btnTelefonNoDuzenle.Location = new System.Drawing.Point(503, 42);
527:    this.btnTelefonNoDuzenle.Name = "btnTelefonNoDuzenle";
528:    this.btnTelefonNoDuzenle.Size = new System.Drawing.Size(134, 23);
529:    this.btnTelefonNoDuzenle.TabIndex = 20;
530:    this.btnTelefonNoDuzenle.Text = "Telefon No Düzenle";
531:    this.btnTelefonNoDuzenle.UseVisualStyleBackColor = true;
532:    this.btnTelefonNoDuzenle.Click += new
System.EventHandler(this.btnTelefonNoDuzenle_Click);
533:    //
534:    // btnMetinSifrele
535:    //
536:    this.btnMetinSifrele.Location = new System.Drawing.Point(363, 42);
537:    this.btnMetinSifrele.Name = "btnMetinSifrele";
538:    this.btnMetinSifrele.Size = new System.Drawing.Size(134, 23);

```

```
539:         this.btnMetinSifrele.TabIndex = 19;
540:         this.btnMetinSifrele.Text = "Metin Sifrele";
541:         this.btnMetinSifrele.UseVisualStyleBackColor = true;
542:         this.btnMetinSifrele.Click += new System.EventHandler(this.btnMetinSifrele_Click);
543:         //
544:         // label15
545:         //
546:         this.label15.AutoSize = true;
547:         this.label15.Dock = System.Windows.Forms.DockStyle.Top;
548:         this.label15.Location = new System.Drawing.Point(0, 0);
549:         this.label15.Name = "label15";
550:         this.label15.Size = new System.Drawing.Size(73, 13);
551:         this.label15.TabIndex = 18;
552:         this.label15.Text = "Metin Islemleri";
553:         this.label15.TextAlign = System.Drawing.ContentAlignment.TopCenter;
554:         //
555:         // btnTürkçeKarakterDüzeltil
556:         //
557:         this.btnTürkçeKarakterDüzeltil.Location = new System.Drawing.Point(223, 42);
558:         this.btnTürkçeKarakterDüzeltil.Name = "btnTürkçeKarakterDüzeltil";
559:         this.btnTürkçeKarakterDüzeltil.Size = new System.Drawing.Size(134, 23);
560:         this.btnTürkçeKarakterDüzeltil.TabIndex = 15;
561:         this.btnTürkçeKarakterDüzeltil.Text = "Türkçe Karakter Düzelt";
562:         this.btnTürkçeKarakterDüzeltil.UseVisualStyleBackColor = true;
563:         this.btnTürkçeKarakterDüzeltil.Click += new
System.EventHandler(this.btnTürkçeKarakterDüzeltil_Click);
564:         //
565:         // btnİlkHarfiBuyukYap
566:         //
567:         this.btnİlkHarfiBuyukYap.Location = new System.Drawing.Point(131, 42);
568:         this.btnİlkHarfiBuyukYap.Name = "btnİlkHarfiBuyukYap";
569:         this.btnİlkHarfiBuyukYap.Size = new System.Drawing.Size(86, 23);
570:         this.btnİlkHarfiBuyukYap.TabIndex = 14;
571:         this.btnİlkHarfiBuyukYap.Text = "İlk Harf Büyük";
572:         this.btnİlkHarfiBuyukYap.UseVisualStyleBackColor = true;
573:         this.btnİlkHarfiBuyukYap.Click += new
System.EventHandler(this.btnİlkHarfiBuyukYap_Click);
574:         //
575:         // label16
576:         //
577:         this.label16.AutoSize = true;
578:         this.label16.Location = new System.Drawing.Point(18, 78);
579:         this.label16.Name = "label16";
580:         this.label16.Size = new System.Drawing.Size(38, 13);
581:         this.label16.TabIndex = 13;
582:         this.label16.Text = "Sonuç";
583:         //
584:         // lbMetinSonuc
585:         //
586:         this.lbMetinSonuc.FormattingEnabled = true;
587:         this.lbMetinSonuc.HorizontalScrollbar = true;
588:         this.lbMetinSonuc.Location = new System.Drawing.Point(62, 78);
589:         this.lbMetinSonuc.Name = "lbMetinSonuc";
590:         this.lbMetinSonuc.Size = new System.Drawing.Size(652, 82);
591:         this.lbMetinSonuc.TabIndex = 12;
592:         //
593:         // btnİlkHarfleriBuyukYap
594:         //
595:         this.btnİlkHarfleriBuyukYap.Location = new System.Drawing.Point(21, 42);
596:         this.btnİlkHarfleriBuyukYap.Name = "btnİlkHarfleriBuyukYap";
597:         this.btnİlkHarfleriBuyukYap.Size = new System.Drawing.Size(104, 23);
598:         this.btnİlkHarfleriBuyukYap.TabIndex = 11;
599:         this.btnİlkHarfleriBuyukYap.Text = "İlk Harfler Büyük";
600:         this.btnİlkHarfleriBuyukYap.UseVisualStyleBackColor = true;
601:         this.btnİlkHarfleriBuyukYap.Click += new
System.EventHandler(this.btnİlkHarfleriBuyukYap_Click);
602:         //
603:         // txtMetin
604:         //
```

```

605:         this.txtMetin.Location = new System.Drawing.Point(119, 12);
606:         this.txtMetin.Name = "txtMetin";
607:         this.txtMetin.Size = new System.Drawing.Size(595, 20);
608:         this.txtMetin.TabIndex = 4;
609:         //
610:         // label17
611:         //
612:         this.label17.AutoSize = true;
613:         this.label17.Location = new System.Drawing.Point(17, 19);
614:         this.label17.Name = "label17";
615:         this.label17.Size = new System.Drawing.Size(33, 13);
616:         this.label17.TabIndex = 3;
617:         this.label17.Text = "Metin";
618:         //
619:         // FrmAnaSayfa
620:         //
621:         this.AutoScaleDimensions = new System.Drawing.SizeF(6F, 13F);
622:         this.AutoScaleMode = System.Windows.Forms.AutoScaleMode.Font;
623:         this.AutoSizeMode = System.Windows.Forms.AutoSizeMode.GrowAndShrink;
624:         this.ClientSize = new System.Drawing.Size(800, 686);
625:         this.Controls.Add(this.pnlMetinIslemleri);
626:         this.Controls.Add(this.pnlKisiIslemleri);
627:         this.Controls.Add(this.pnlSayiIslemleri);
628:         this.Controls.Add(this.pnlSifreIslemleri);
629:         this.Controls.Add(this.pnlTarihIslemleri);
630:         this.Controls.Add(this.label1);
631:         this.Controls.Add(this.cmbIslemTuru);
632:         this.Name = "FrmAnaSayfa";
633:         this.StartPosition = System.Windows.Forms.FormStartPosition.CenterScreen;
634:         this.Text = "Islemler Ekrani";
635:         this.Load += new System.EventHandler(this.FrmAnaSayfa_Load);
636:         this.pnlTarihIslemleri.ResumeLayout(false);
637:         this.pnlTarihIslemleri.PerformLayout();
638:         this.pnlSifreIslemleri.ResumeLayout(false);
639:         this.pnlSifreIslemleri.PerformLayout();
640:         this.pnlSayiIslemleri.ResumeLayout(false);
641:         this.pnlSayiIslemleri.PerformLayout();
642:         this.pnlKisiIslemleri.ResumeLayout(false);
643:         this.pnlKisiIslemleri.PerformLayout();
644:         this.pnlMetinIslemleri.ResumeLayout(false);
645:         this.pnlMetinIslemleri.PerformLayout();
646:         this.ResumeLayout(false);
647:         this.PerformLayout();
648:
649:     }

```

1.1.3.2 Program Class

Class Hierarchy

CSharpHelperMethods.Program

C#

```
internal static class Program;
```

File

Program.cs (see page 93)

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 63)	This is file CSharpHelperMethods.csproj.
CSharpHelperMethods.sln (see page 63)	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 77)	This is file FrmAnaSayfa.Designer.cs.
Kisilslemleri.cs (see page 88)	This is file Kisilslemleri.cs.
MetinIslemleri.cs (see page 90)	This is file MetinIslemleri.cs.
Program.cs (see page 93)	This is file Program.cs.
Resources.Designer.cs (see page 93)	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 95)	This is file SayIslemleri.cs.
Settings.Designer.cs (see page 97)	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 98)	This is file Sifrelemelslemleri.cs.
Siralamalslemleri.cs (see page 100)	This is file Siralamalslemleri.cs.
TarihIslemleri.cs (see page 102)	This is file TarihIslemleri.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-43b2efal8625")]
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 lazım olan DataTable vb. verilerin düzenlenmesini saglayan
10:    siniflardir.
11:    /// </summary>
12:    public static class DataTableIslemleri
13:    {
14:        /// <summary>
15:        /// Parametre olarak girilen herhangi bir sinif türünden listeyi
16:        /// Idael bir sekilde Nullable alan içermeyen bir DataTable'a çevirmek için
17:        /// kullanılan metottur.
18:        /// </summary>
19:        /// <typeparam name="T">Generic Nesne Tipi</typeparam>
20:        /// <param name="liste">Liste Bilgisi</param>
21:        /// <returns>DataTable cinsinde Nesne</returns>
22:        public static DataTable ToDataTable<T>(List<T> liste)
23:        {
24:            try
25:            {
26:                var sonuc = new DataTable(typeof(T).Name);
27:
28:                var ozellikler = typeof(T).GetFields(BindingFlags.Public |
29:                BindingFlags.CreateInstance | BindingFlags.Default | BindingFlags.DeclaredOnly |
30:                BindingFlags.ExactBinding | BindingFlags.FlattenHierarchy | BindingFlags.IgnoreReturn |
31:                BindingFlags.Instance | BindingFlags.InvokeMethod | BindingFlags.NonPublic |
32:                BindingFlags.OptionalParamBinding | BindingFlags.SuppressChangeType);
33:
34:                var ozellikler2 = typeof(T).GetProperties(BindingFlags.Public |
35:                BindingFlags.CreateInstance | BindingFlags.Default | BindingFlags.DeclaredOnly |
36:                BindingFlags.ExactBinding | BindingFlags.FlattenHierarchy | BindingFlags.IgnoreReturn |
37:                BindingFlags.Instance | BindingFlags.InvokeMethod | BindingFlags.NonPublic |
38:                BindingFlags.OptionalParamBinding | BindingFlags.SuppressChangeType);
39:
40:                foreach (var ozellik in ozellikler)
41:                {
42:                    var varMi = false;
43:                    foreach (var ozellik2 in ozellikler2)
44:                    {
45:                        if (!ozellik.Name.Contains(ozellik2.Name)) continue;
46:                        varMi = true;
47:                        var ozellikAdi = ozellik.Name.Replace("<", "");
48:                        var ozellikDizi = ozellikAdi.Split('>');
49:                        var ozellikTemp = ozellikDizi[0].Trim();
50:                        //ozellikTemp = ozellikTemp.Substring(0, ozellikTemp.Length -
51:                    1);
52:
53:                        if (ozellikTemp == ozellik2.Name)
54:                        {
55:                            if (sonuc.Columns.IndexOf(new DataColumn(ozellik2.Name))
56:                                >= 0) continue;
57:
58:                            sonuc.Columns.Add(ozellik2.Name,
59:                                BaseType(ozellik2.PropertyType));
60:
61:                            break;
62:                        }
63:                    }
64:                }
65:            }
66:            catch { }
67:        }
68:    }
69: }

```

```

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:     return sonuc;
85: }
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 uzantileri ile ilgili yardımcı sınıf
8:     /// </summary>
9:     public static class DosyaIslemleri
10:    {
11:        /// <summary>
12:        /// Dosya uzantisinden dosyanın türünün ne olduğunu getiren metottur
13:        /// </summary>
14:        /// <param name="extension">Dosya Uzantisi</param>
15:        /// <returns>Dosya Türü Bilgisi</returns>
16:        public static string GetirIcerikTipiDosyaUzantisiIle(string extension)
17:        {
18:            var contentType = String.Empty;
19:            extension = extension.Replace(".", "").ToLower();
20:            switch (extension)
21:            {
22:                case "pdf": contentType = "application/pdf"; break;
23:                case "jpeg": contentType = "image/jpeg"; break;
24:                case "jpg": contentType = "image/jpg"; break;
25:                case "png": contentType = "image/png"; break;
26:                case "bmp": contentType = "image/bmp"; break;
27:                case "doc": contentType = "application/msword"; break;
28:                case "pptx": contentType =
29:                    "application/vnd.openxmlformats-officedocument.presentationml.presentation"; break;
30:                case "xls": contentType = "application/vnd.ms-excel"; break;
31:                case "txt": contentType = "text/plain"; break;
32:                case "xlsx": contentType =
33:                    "application/vnd.openxmlformats-officedocument.spreadsheetml.sheet"; break;
34:                case "rtf": contentType = "application/rtf"; break;
35:                case "docx": contentType =
36:                    "application/vnd.openxmlformats-officedocument.wordprocessingml.document"; break;
37:                case "ppt": contentType = "application/vnd.ms-powerpoint"; break;
38:                case "zip": contentType = "application/zip"; break;
39:                case "rar": contentType = "application/x-rar-compressed"; break;
40:                case "7z": contentType = "application/x-7z-compressed"; break;
41:                default: contentType = "application/octet-stream"; break;
42:            }
43:
44:            return contentType;
45:        }
46:
47:        /// <summary>
48:        /// Dosyanın türünden dosyanın uzantisinin ne olduğunu getiren metottur
49:        /// </summary>
50:        /// <param name="mimeType">Mime Type</param>
51:        /// <returns>Dosya Türü Bilgisi</returns>
52:        public static string GetirDosyaUzantisiMimeTypeIle(string mimeType)
53:        {
54:            string dosyaUzantisi;
55:            mimeType = mimeType.ToLower();
56:            switch (mimeType)
57:            {
58:                case "application/pdf": dosyaUzantisi = "pdf"; break;
59:                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 dosyanın belirlenen kriterlere uyup uymadığını kontrol
eden metottur.
79:     /// Dosyanın uzantisi istediğimiz formatta değilse yanlış olarak geri
döndürecektir.s
80:     /// </summary>
81:     /// <param name="extension">Dosya Uzantisi</param>
82:     /// <returns>Eklenenecek Dosyanın Uygun Olup Olmadığı 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 dosyanın başka bir program tarafından o anda
kullanımda olup olmadığının kontrol edildiği bir metottur.
92:     /// </summary>
93:     /// <param name="dosya">Dosya Bilgisi</param>
94:     /// <returns>Evet veya Hayır Bilgisi</returns>
95:     public static bool DosyaKilitliMi(FileInfo dosya)
96:     {
97:         FileStream stream = null;
98:         try
99:         {
100:             stream = dosya.Open(FileMode.Open, FileAccess.ReadWrite,
FileShare.None);
101:         }
102:         catch (IOException)
103:         {
104:             // eger dosya başka bir uygulama tarafından kullanılıyorsa
105:             // burada true dönüyor ki
106:             // kullanıcı bu metottan gelen bilgiye göre davranışın
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 CSharpHelperMethods.YardimciSiniflar;
2: using System.Text;
3: using System.Windows.Forms;
4:
5: namespace CSharpHelperMethods
6: {
7:     public partial class FrmAnaSayfa : Form
8:     {
9:         public FrmAnaSayfa()
10:        {
11:            InitializeComponent();
12:        }
13:
14:        #region Formun Yüklmesi ve Panel Seçimi
15:        /// <summary>
16:        /// Form İlk Yüklendiği zaman ekrandaki bütün panellerin erisilemez hale
17:        /// Bu metotta, çünkü hangi panelin aktifleseceğinin
18:        /// Combobox üzerinden seçilmesi lazım
19:        /// </summary>
20:        /// <param name="sender"></param>
21:        /// <param name="e"></param>
22:        private void FrmAnaSayfa_Load(object sender, System.EventArgs e)
23:        {
24:            cmbIslemTuru.DropDownStyle = ComboBoxStyle.DropDownList;
25:            #region Sayfa İlk Yüklendiğinde Tüm Panellerin Erisime Kapanması
26:            pnlTarihIslemleri.Enabled = false;
27:            pnlSifreIslemleri.Enabled = false;
28:            pnlSayiIslemleri.Enabled = false;
29:            pnlKisiIslemleri.Enabled = false;
30:            pnlMetinIslemleri.Enabled = false;
31:            #endregion

```

```

32:     }
33:
34:     /// <summary>
35:     /// Islem Türü Combobox'ında seçilen degere göre
36:     /// Ekrandaki panellerden ilgili olan haricinde hepsinin erisilemez hale
getirilmesi
37:     /// Eger bir sey seçilmemisse, tamaminin erisilemez hale getirilmesi için
38:     /// Gerekli islemlerin gerçekleştirildiği metottur.
39:     /// </summary>
40:     /// <param name="sender"></param>
41:     /// <param name="e"></param>
42:     private void cmbIslemTuru_SelectedIndexChanged(object sender, System.EventArgs
e)
43:     {
44:         #region Combobox'tan Seçilen Degere Göre İlgili Paneli Aktif Hale Getirme
45:         switch (cmbIslemTuru.SelectedIndex)
46:         {
47:             // Metin Islemleri
48:             case 0:
49:             {
50:                 pnlMetinIslemleri.Enabled = true;
51:                 pnlTarihIslemleri.Enabled = false;
52:                 pnlSifreIslemleri.Enabled = false;
53:                 pnlSayiIslemleri.Enabled = false;
54:                 pnlKisiIslemleri.Enabled = false;
55:                 break;
56:             }
57:             // Tarih Islemleri
58:             case 1:
59:             {
60:                 pnlMetinIslemleri.Enabled = false;
61:                 pnlTarihIslemleri.Enabled = true;
62:                 pnlSifreIslemleri.Enabled = false;
63:                 pnlSayiIslemleri.Enabled = false;
64:                 pnlKisiIslemleri.Enabled = false;
65:                 break;
66:             }
67:             // Sayi Islemleri
68:             case 2:
69:             {
70:                 pnlMetinIslemleri.Enabled = false;
71:                 pnlTarihIslemleri.Enabled = false;
72:                 pnlSifreIslemleri.Enabled = false;
73:                 pnlSayiIslemleri.Enabled = true;
74:                 pnlKisiIslemleri.Enabled = false;
75:                 break;
76:             }
77:             // Sifre Islemleri
78:             case 3:
79:             {
80:                 pnlMetinIslemleri.Enabled = false;
81:                 pnlTarihIslemleri.Enabled = false;
82:                 pnlSifreIslemleri.Enabled = true;
83:                 pnlSayiIslemleri.Enabled = false;
84:                 pnlKisiIslemleri.Enabled = false;
85:                 break;
86:             }
87:             // Kisi Islemleri
88:             case 4:
89:             {
90:                 pnlMetinIslemleri.Enabled = false;
91:                 pnlTarihIslemleri.Enabled = false;
92:                 pnlSifreIslemleri.Enabled = false;
93:                 pnlSayiIslemleri.Enabled = false;
94:                 pnlKisiIslemleri.Enabled = true;
95:                 break;
96:             }
97:             // Bunlar haricinde herhangi bir sey
98:             default:

```



```

99:         {
100:             pnlMetinIslemleri.Enabled = false;
101:             pnlTarihIslemleri.Enabled = false;
102:             pnlSifreIslemleri.Enabled = false;
103:             pnlSayiIslemleri.Enabled = false;
104:             pnlKisiIslemleri.Enabled = false;
105:             break;
106:         }
107:     }
108: #endregion
109: }
110: #endregion
111:
112: #region Tarih Islemleri
113:
114:     /// <summary>
115:     /// Paneldeki Baslangiç Tarihi seçili ve tarih girilmissse, o tarihten bugüne
116:     kaç yas oldugunun hesaplandigi metottur. Ilgili butona tiklandigi zaman çalisir
117:     /// </summary>
118:     /// <param name="sender"></param>
119:     /// <param name="e"></param>
120:     private void btnYasHesapla_Click(object sender, System.EventArgs e)
121:     {
122:         if (!dtpBaslangicTarihi.Checked)
123:             MessageBox.Show("Baslangiç Tarihi girilmedigi için bu islem
124: gerçeklestirilemez");
125:         else
126:         {
127:             lbSonuc.Items.Clear();
128:             var sb = new StringBuilder();
129:             sb.Append("Dogum tarihi seçilen kisi ");
130:             sb.Append(TarihIslemleri.YasHesapla(dtpBaslangicTarihi.Value));
131:             sb.Append(" yasindadir");
132:             lbSonuc.Items.Add(sb.ToString());
133:         }
134:     }
135:     /// <summary>
136:     /// Paneldeki Baslangiç Tarihi seçili ve tarih girilmissse, o tarihten bugüne
137:     kaç yas oldugunun yıl, ay ve gün olarak metinsel bir sekilde hesaplandigi metottur.
138:     /// Ilgili butona tiklandigi zaman çalisir
139:     /// </summary>
140:     /// <param name="sender"></param>
141:     /// <param name="e"></param>
142:     private void btnYasHesaplaMetinsel_Click(object sender, System.EventArgs e)
143:     {
144:         if (!dtpBaslangicTarihi.Checked)
145:             MessageBox.Show("Baslangiç Tarihi girilmedigi için bu islem
146: gerçeklestirilemez");
147:         else
148:         {
149:             lbSonuc.Items.Clear();
150:             var sb = new StringBuilder();
151:             sb.Append(TarihIslemleri.YasHesaplaMetinsel(dtpBaslangicTarihi.Value));
152:             lbSonuc.Items.Add(sb.ToString());
153:         }
154:     }
155:     /// <summary>
156:     /// Paneldeki Baslangiç Tarihi ve Bitis Tarihi alanlari seçili ve tarih
157:     girilmissse,
158:     /// Söz konusu tarihler arasında kaç yıl, kaç ay ve gün oldugunun bilgisinin
159:     hazırlandigi metottur.
160:     /// Ilgili butona tiklandigi zaman çalisir
161:     /// </summary>
162:     /// <param name="sender"></param>
163:     /// <param name="e"></param>
164:     private void btnTarihAraligi_Click(object sender, System.EventArgs e)
165:     {

```

```

162:         if (!dtpBaslangicTarihi.Checked || !dtpBitisTarihi.Checked)
163:             MessageBox.Show("Baslangiç Tarihi veya Bitis Tarihi girilmediği için
bu işlem gerçekleştirilemez");
164:         else if (dtpBaslangicTarihi.Value.Date > dtpBitisTarihi.Value.Date)
165:             MessageBox.Show("Baslangiç Tarihi Bitis Tarihinden büyük olamaz");
166:         else
167:         {
168:             lbSonuc.Items.Clear();
169:             var sb = new StringBuilder();
170:             sb.Append("İki tarih arası: ");
171:             sb.Append(TarihIslemleri.TarihAraligiHesapla(dtpBaslangicTarihi.Value.Date,
dtpBitisTarihi.Value.Date));
172:             lbSonuc.Items.Add(sb.ToString());
173:         }
174:     }
175:
176:     /// <summary>
177:     /// Paneldeki Baslangiç Tarihi ve Bitis Tarihi alanları seçili ve tarih
girilmisse,
178:     /// Söz konusu tarihler arasında kaç ay olduğunun bilgisinin hazırlandığı
metottur.
179:     /// İlgili butona tıklandığı zaman çalışır
180:     /// </summary>
181:     /// <param name="sender"></param>
182:     /// <param name="e"></param>
183:     private void btnAyFarki_Click(object sender, System.EventArgs e)
184:     {
185:         if (!dtpBaslangicTarihi.Checked || !dtpBitisTarihi.Checked)
186:             MessageBox.Show("Baslangiç Tarihi veya Bitis Tarihi girilmediği için
bu işlem gerçekleştirilemez");
187:         else if (dtpBaslangicTarihi.Value.Date > dtpBitisTarihi.Value.Date)
188:             MessageBox.Show("Baslangiç Tarihi Bitis Tarihinden büyük olamaz");
189:         else
190:         {
191:             lbSonuc.Items.Clear();
192:             var sb = new StringBuilder();
193:             sb.Append("Ay Farkı: ");
194:             sb.Append(TarihIslemleri.AyFarki(dtpBaslangicTarihi.Value.Date,
dtpBitisTarihi.Value.Date));
195:             lbSonuc.Items.Add(sb.ToString());
196:         }
197:     }
198: #endregion
199:
200: #region Sifre Islemleri
201:     /// <summary>
202:     /// Paneldeki metin kutusu dolu ise
203:     /// Girilen metnin MD5 algoritması ile şifrelenmiş halini
204:     /// Listbox üzerine yazdıran metottur.
205:     /// </summary>
206:     /// <param name="sender"></param>
207:     /// <param name="e"></param>
208:     private void btnMd5Hash_Click(object sender, System.EventArgs e)
209:     {
210:         if (string.IsNullOrEmpty(txtSifrelenecekMetin.Text.Trim()))
211:             MessageBox.Show("Şifrelenecek metin girilmediği için bu işlem
gerçekleştirilemez");
212:         else
213:         {
214:             lbSifreSonuc.Items.Clear();
215:             var sb = new StringBuilder();
216:             sb.Append("Girilen Metnin MD5 Karşılığı: ");
217:             sb.Append(SifrelemeIslemleri.HesaplaMD5(txtSifrelenecekMetin.Text.Trim()));
218:             lbSifreSonuc.Items.Add(sb.ToString());
219:         }
220:     }
221:

```

```

222:         /// <summary>
223:         /// Paneldeki metin kutusu dolu ise
224:         /// Girilen metnin SHA 256 algoritmasi ile sifrelenmis halini
225:         /// Listbox üzerine yazdiran metottur.
226:         /// </summary>
227:         /// <param name="sender"></param>
228:         /// <param name="e"></param>
229:         private void btnSha256Hash_Click(object sender, System.EventArgs e)
230:         {
231:             if (string.IsNullOrEmpty(txtSifrelenecekMetin.Text.Trim()))
232:                 MessageBox.Show("Sifrelenecek metin girilmedigi için bu islem
gerçeklestirilemez");
233:             else
234:             {
235:                 lbSifreSonuc.Items.Clear();
236:                 var sb = new StringBuilder();
237:                 sb.Append("Girilen Metnin SHA 256 Karsiligi: ");
238:                 sb.Append(SifrelemeIslemleri.HesaplaSHA256(txtSifrelenecekMetin.Text.Trim()));
239:                 lbSifreSonuc.Items.Add(sb.ToString());
240:             }
241:         }
242:
243:         /// <summary>
244:         /// Paneldeki metin kutusu dolu ise
245:         /// Girilen metnin SHA 512 algoritmasi ile sifrelenmis halini
246:         /// Listbox üzerine yazdiran metottur.
247:         /// </summary>
248:         /// <param name="sender"></param>
249:         /// <param name="e"></param>
250:         private void btnSha512Hash_Click(object sender, System.EventArgs e)
251:         {
252:             if (string.IsNullOrEmpty(txtSifrelenecekMetin.Text.Trim()))
253:                 MessageBox.Show("Sifrelenecek metin girilmedigi için bu islem
gerçeklestirilemez");
254:             else
255:             {
256:                 lbSifreSonuc.Items.Clear();
257:                 var sb = new StringBuilder();
258:                 sb.Append("Girilen Metnin SHA 512 Karsiligi: ");
259:                 sb.Append(SifrelemeIslemleri.HesaplaSHA512(txtSifrelenecekMetin.Text.Trim()));
260:                 lbSifreSonuc.Items.Add(sb.ToString());
261:             }
262:         }
263:
264:         /// <summary>
265:         /// Ilgili butona tiklandiginda rastgele sifre üretilmesini saglayan metottur
266:         /// </summary>
267:         /// <param name="sender"></param>
268:         /// <param name="e"></param>
269:         private void btnRastgeleSifreUret_Click(object sender, System.EventArgs e)
270:         {
271:             lbSifreSonuc.Items.Clear();
272:             var sb = new StringBuilder();
273:             sb.Append("Rastgele Üretilen Sifre: ");
274:             sb.Append(SifrelemeIslemleri.RastgeleSifreUret());
275:             lbSifreSonuc.Items.Add(sb.ToString());
276:         }
277:
278:         /// <summary>
279:         /// Paneldeki metin kutusu dolu ise
280:         /// Burada yazilan metnin bir sifre olarak belirli kriterlere uygun olup
olmadiginin
281:         /// Gösterilmesini saglayan bir metottur
282:         /// </summary>
283:         /// <param name="sender"></param>
284:         /// <param name="e"></param>
285:         private void btnSifreUygunMu_Click(object sender, System.EventArgs e)

```

```

286:         {
287:             if (string.IsNullOrEmpty(txtSifrelenecekMetin.Text.Trim()))
288:                 MessageBox.Show("Sifre metni girilmediği için bu işlem
gerçekleştirilemez");
289:             else
290:             {
291:                 lbSifreSonuc.Items.Clear();
292:                 var sb = new StringBuilder();
293:                 sb.Append("Girilen Sifrenin Uygunluk Durumu: ");
294:                 sb.Append(SifrelemeIslemleri.SifreUygunMu(txtSifrelenecekMetin.Text.Trim()));
295:                 lbSifreSonuc.Items.Add(sb.ToString());
296:             }
297:         }
298:     #endregion
299:
300:     #region Sayi Islemleri
301:     /// <summary>
302:     /// Paneldeki metin kutusuna girilen metnin sayısal bir değer olup olmadığını
303:     /// kontrol eden ve sonucu gösteren metottur.
304:     /// </summary>
305:     /// <param name="sender"></param>
306:     /// <param name="e"></param>
307:     private void btnSayısalMi_Click(object sender, System.EventArgs e)
308:     {
309:         if (string.IsNullOrEmpty(txtSayi.Text.Trim()))
310:             MessageBox.Show("Metin girilmediği için bu işlem gerçekleştirilemez");
311:         else
312:         {
313:             lbSayiSonuc.Items.Clear();
314:             var sb = new StringBuilder();
315:             sb.Append("Girilen Metin Sayı mı?: ");
316:             sb.Append(SayıIslemleri.SayısalMi(txtSayi.Text.Trim()) ? "Evet" :
"Hayır");
317:             lbSayiSonuc.Items.Add(sb.ToString());
318:         }
319:     }
320:
321:     /// <summary>
322:     /// Paneldeki metin kutusuna girilen para birimi cinsinden rakamın
323:     /// yazıya çevrilmesini sağlayan ve sonucu listbox üzerinde gösteren bir
metottur.
324:     /// </summary>
325:     /// <param name="sender"></param>
326:     /// <param name="e"></param>
327:     private void btnMetneDonustur_Click(object sender, System.EventArgs e)
328:     {
329:         if (string.IsNullOrEmpty(txtSayi.Text.Trim()))
330:             MessageBox.Show("Metin girilmediği için bu işlem gerçekleştirilemez");
331:         else if (!SayıIslemleri.SayısalMi(txtSayi.Text.Trim().Replace(".", "")))
332:             MessageBox.Show("Girilen metin sayısal bir değer olmadığı için bu
işlem gerçekleştirilemez");
333:         else
334:         {
335:             var metinSadeceSayıKismi = txtSayi.Text.Trim();
336:             var sayi = SayıIslemleri.SayıHaliCevir(metinSadeceSayıKismi);
337:             lbSayiSonuc.Items.Clear();
338:             var sb = new StringBuilder();
339:             sb.Append("Girilen Sayının Metin Hali (Parasal): ");
340:             sb.Append(SayıIslemleri.SayıYazıyaCevir(sayi));
341:             lbSayiSonuc.Items.Add(sb.ToString());
342:         }
343:     }
344:
345:     /// <summary>
346:     /// Rastgele olacak şekilde 6 haneli kod oluşturmaya sağlayan bir metottur.
347:     /// </summary>
348:     /// <param name="sender"></param>

```

```

349:         /// <param name="e"></param>
350:     private void btnOnayKoduOlustur_Click(object sender, System.EventArgs e)
351:     {
352:         lbSayiSonuc.Items.Clear();
353:         var sb = new StringBuilder();
354:         sb.Append("Olusturulan 6 haneli onay kodu: ");
355:         sb.Append(SayiIslemleri.AltiHaneliOnayKoduOlustur());
356:         lbSayiSonuc.Items.Add(sb.ToString());
357:     }
358: #endregion
359:
360: #region Kisi Islemleri
361:     /// <summary>
362:     /// Paneldeki metin kutusuna girilen TC Kimlik numarasinin
363:     /// Standart TC Kimlik No algoritmasina uygun olup olmadigini kontrol eden bir
364:     /// </summary>
365:     /// <param name="sender"></param>
366:     /// <param name="e"></param>
367:     private void btnTcKimlikDogrula_Click(object sender, System.EventArgs e)
368:     {
369:         if (string.IsNullOrEmpty(txtKisiMetni.Text.Trim()))
370:             MessageBox.Show("Metin girilmedigi için bu islem gerçekleştirilemez");
371:         else if (!SayiIslemleri.SayisalMi(txtKisiMetni.Text.Trim()))
372:             MessageBox.Show("Girilen metin sayisal bir deger olmadigi için bu
373: islem gerçekleştirilemez");
374:         else if (txtKisiMetni.Text.Trim().Length != 11)
375:             MessageBox.Show("Girilen metin 11 karakter olmadigi için bu islem
376: gerçekleştirilemez");
377:         else
378:         {
379:             lbKisiSonuc.Items.Clear();
380:             var sb = new StringBuilder();
381:             sb.Append("TC Kimlik No Dogru Mu?: ");
382:             sb.Append(KisiIslemleri.DogrulaTcKimlikNo(txtKisiMetni.Text.Trim()) ?
383: "Evet" : "Hayir");
384:             lbKisiSonuc.Items.Add(sb.ToString());
385:         }
386:     }
387: #endregion
388:
389: /// <summary>
390: /// Paneldeki metin kutusuna girilen IBAN Numarasinin
391: /// Standart IBAN algoritmasina uygun olup olmadigini kontrol eden bir
392: /// </summary>
393: /// <param name="sender"></param>
394: /// <param name="e"></param>
395:     private void btnIbanDogrula_Click(object sender, System.EventArgs e)
396:     {
397:         if (string.IsNullOrEmpty(txtKisiMetni.Text.Trim()))
398:             MessageBox.Show("Metin girilmedigi için bu islem gerçekleştirilemez");
399:         else if (txtKisiMetni.Text.Trim().Length != 26)
400:             MessageBox.Show("Girilen metin 26 karakter olmadigi için bu islem
401: gerçekleştirilemez. Karakter sayisi: " + txtKisiMetni.Text.Trim().Length);
402:         else
403:         {
404:             lbKisiSonuc.Items.Clear();
405:             var sb = new StringBuilder();
406:             sb.Append("IBAN Dogru Mu?: ");
407:             sb.Append(KisiIslemleri.DogrulaIBAN(txtKisiMetni.Text.Trim()) ? "Evet"
408: : "Hayir");
409:             lbKisiSonuc.Items.Add(sb.ToString());
410:         }
411:     }
412: #endregion
413:
414: /// <summary>
415: /// Paneldeki metin kutusuna girilen metnin
416: /// Geçerli bir e-posta adresi olup olmadigini kontrol eden bir metottur.
417: /// </summary>

```

```

411:         /// <param name="sender"></param>
412:         /// <param name="e"></param>
413:         private void btnEPostaDogrula_Click(object sender, System.EventArgs e)
414:         {
415:             if (string.IsNullOrEmpty(txtKisiMetni.Text.Trim()))
416:                 MessageBox.Show("Metin girilmedigi için bu islem gerçekleştirilemez");
417:             else
418:             {
419:                 lbKisiSonuc.Items.Clear();
420:                 var sb = new StringBuilder();
421:                 sb.Append("E-Posta Dogru Mu?: ");
422:                 sb.Append(KisiIslemleri.DogrulaEPosta(txtKisiMetni.Text.Trim()) ?
"Ev" : "Hayir");
423:                 lbKisiSonuc.Items.Add(sb.ToString());
424:             }
425:         }
426:     #endregion
427:
428:     #region Metin Islemleri
429:     /// <summary>
430:     /// Paneldeki metin kutusuna girilen metnin
431:     /// Kelimelerinin ilk harflerini büyük yapmaya yarayan bir metottur.
432:     /// </summary>
433:     /// <param name="sender"></param>
434:     /// <param name="e"></param>
435:     private void btnIlkHarfleriBuyukYap_Click(object sender, System.EventArgs e)
436:     {
437:         if (string.IsNullOrEmpty(txtMetin.Text.Trim()))
438:             MessageBox.Show("Metin girilmedigi için bu islem gerçekleştirilemez");
439:         else
440:         {
441:             lbMetinSonuc.Items.Clear();
442:             var sb = new StringBuilder();
443:             sb.Append("Metnin Düzenlenmis Hali: ");
444:             sb.Append(txtMetin.Text.Trim().MetinIlkKarakterleriBuyukYap());
445:             lbMetinSonuc.Items.Add(sb.ToString());
446:         }
447:     }
448:
449:     /// <summary>
450:     /// Paneldeki metin kutusuna girilen metnin
451:     /// Sadece ilk harfini büyük yapmaya yarayan bir metottur.
452:     /// </summary>
453:     /// <param name="sender"></param>
454:     /// <param name="e"></param>
455:     private void btnIlkHarfiBuyukYap_Click(object sender, System.EventArgs e)
456:     {
457:         if (string.IsNullOrEmpty(txtMetin.Text.Trim()))
458:             MessageBox.Show("Metin girilmedigi için bu islem gerçekleştirilemez");
459:         else
460:         {
461:             lbMetinSonuc.Items.Clear();
462:             var sb = new StringBuilder();
463:             sb.Append("Metnin Düzenlenmis Hali: ");
464:
sb.Append(MetinIslemleri.MetinIlkKarakterBuyukDigerleriKucukYap(txtMetin.Text.Trim()));
465:             lbMetinSonuc.Items.Add(sb.ToString());
466:         }
467:     }
468:
469:     /// <summary>
470:     /// Paneldeki metin kutusuna girilen metnin
471:     /// İçerisindeki Türkçe karakterleri İngilizce hallerine çeviren bir metottur.
472:     /// </summary>
473:     /// <param name="sender"></param>
474:     /// <param name="e"></param>
475:     private void btnTurkceKarakterDuzelt_Click(object sender, System.EventArgs e)
476:     {
477:         if (string.IsNullOrEmpty(txtMetin.Text.Trim()))

```

```

478:         MessageBox.Show("Metin girilmediği için bu işlem gerçekleştirilemez");
479:     else
480:     {
481:         lbMetinSonuc.Items.Clear();
482:         var sb = new StringBuilder();
483:         sb.Append("Metnin Düzenlenmiş Hali: ");
484:
sb.Append(MetinIslemleri.TurkceKarakterleriDuzelt(txtMetin.Text.Trim()));
485:         lbMetinSonuc.Items.Add(sb.ToString());
486:     }
487: }
488:
489:     /// <summary>
490:     /// KVKK durumlarından dolayı TC Kimlik No ve Vergi No gibi alanların
491:     /// Matbu çıktılarda gizlenmesi gibi bir durum söz konusu olduğundan
492:     /// Girilen bu 10 veya 11 karakterlik metinleri şifrelemeye yarayan bir
metottur
493:     /// </summary>
494:     /// <param name="sender"></param>
495:     /// <param name="e"></param>
496:     private void btnMetinSifrele_Click(object sender, System.EventArgs e)
497:     {
498:         if (string.IsNullOrEmpty(txtMetin.Text.Trim()))
499:             MessageBox.Show("Metin girilmediği için bu işlem gerçekleştirilemez");
500:         else if (!SayiIslemleri.SayisalMi(txtMetin.Text.Trim()))
501:             MessageBox.Show("Girilen metin sayısal bir değer olmadığı için bu
işlem gerçekleştirilemez");
502:         else if (txtMetin.Text.Trim().Length != 11 && txtMetin.Text.Trim().Length
!= 10)
503:             MessageBox.Show("Girilen metin 10 veya 11 karakter olmadığı için bu
işlem gerçekleştirilemez");
504:         else
505:         {
506:             lbMetinSonuc.Items.Clear();
507:             var sb = new StringBuilder();
508:             sb.Append("TC Kimlik No veya Vergi No'nun Şifrelenmiş Hali: ");
509:             sb.Append(MetinIslemleri.MetinSifrele(txtMetin.Text.Trim()));
510:             lbMetinSonuc.Items.Add(sb.ToString());
511:         }
512:     }
513:
514:     /// <summary>
515:     /// Ekrandan çeşitli karakterler içeren telefon numaraları alındığı zaman
516:     /// Bu telefon numarasını olması gerektiği hale çevirmeye yarayan bir metottur.
517:     /// </summary>
518:     /// <param name="sender"></param>
519:     /// <param name="e"></param>
520:     private void btnTelefonNoDuzenle_Click(object sender, System.EventArgs e)
521:     {
522:         if (string.IsNullOrEmpty(txtMetin.Text.Trim()))
523:             MessageBox.Show("Metin girilmediği için bu işlem gerçekleştirilemez");
524:         else
525:         {
526:             lbMetinSonuc.Items.Clear();
527:             var sb = new StringBuilder();
528:             sb.Append("Telefon Numarasının Düzenlenmiş Hali: ");
529:             sb.Append(MetinIslemleri.TelefonNoDuzenle(txtMetin.Text.Trim()));
530:             lbMetinSonuc.Items.Add(sb.ToString());
531:         }
532:     }
533: #endregion
534: }
535: }

```

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.txtSifreleneccekMetin = 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[] {
96:             "Metin Islemleri",
97:             "Tarih Islemleri",
98:             "Sayi Islemleri",
99:             "Sifre Islemleri",
100:             "Kisi Islemleri"});
101:         this.cmbIslemTuru.Location = new System.Drawing.Point(132, 13);
102:         this.cmbIslemTuru.Name = "cmbIslemTuru";
103:         this.cmbIslemTuru.Size = new System.Drawing.Size(200, 21);
104:         this.cmbIslemTuru.TabIndex = 0;
105:         this.cmbIslemTuru.SelectedIndexChanged += new
System.EventHandler(this.cmbIslemTuru_SelectedIndexChanged);
106:         //
107:         // label1
108:         //
109:         this.label1.AutoSize = true;
110:         this.label1.Location = new System.Drawing.Point(12, 16);
111:         this.label1.Name = "label1";
112:         this.label1.Size = new System.Drawing.Size(93, 13);
113:         this.label1.TabIndex = 1;
114:         this.label1.Text = "Islem Türü Seçiniz";
115:         //
116:         // pnlTarihIslemleri
117:         //
118:         this.pnlTarihIslemleri.BorderStyle =
System.Windows.Forms.BorderStyle.FixedSingle;
119:         this.pnlTarihIslemleri.Controls.Add(this.btnAyFarki);
120:         this.pnlTarihIslemleri.Controls.Add(this.label8);
121:         this.pnlTarihIslemleri.Controls.Add(this.btnTarihAraligi);
122:         this.pnlTarihIslemleri.Controls.Add(this.btnYasHesaplaMetinsel);
123:         this.pnlTarihIslemleri.Controls.Add(this.btnYasHesapla);
124:         this.pnlTarihIslemleri.Controls.Add(this.label4);
125:         this.pnlTarihIslemleri.Controls.Add(this.lbSonuc);

```

```

126:         this.pnlTarihIslemleri.Controls.Add(this.dtpBitisTarihi);
127:         this.pnlTarihIslemleri.Controls.Add(this.dtpBaslangicTarihi);
128:         this.pnlTarihIslemleri.Controls.Add(this.label3);
129:         this.pnlTarihIslemleri.Controls.Add(this.label2);
130:         this.pnlTarihIslemleri.Location = new System.Drawing.Point(15, 56);
131:         this.pnlTarihIslemleri.Name = "pnlTarihIslemleri";
132:         this.pnlTarihIslemleri.Size = new System.Drawing.Size(374, 181);
133:         this.pnlTarihIslemleri.TabIndex = 2;
134:         //
135:         // btnAyFarki
136:         //
137:         this.btnAyFarki.Location = new System.Drawing.Point(286, 76);
138:         this.btnAyFarki.Name = "btnAyFarki";
139:         this.btnAyFarki.Size = new System.Drawing.Size(79, 23);
140:         this.btnAyFarki.TabIndex = 12;
141:         this.btnAyFarki.Text = "Ay Farki";
142:         this.btnAyFarki.UseVisualStyleBackColor = true;
143:         this.btnAyFarki.Click += new System.EventHandler(this.btnAyFarki_Click);
144:         //
145:         // label8
146:         //
147:         this.label8.AutoSize = true;
148:         this.label8.Dock = System.Windows.Forms.DockStyle.Top;
149:         this.label8.Location = new System.Drawing.Point(0, 0);
150:         this.label8.Name = "label8";
151:         this.label8.Size = new System.Drawing.Size(71, 13);
152:         this.label8.TabIndex = 11;
153:         this.label8.Text = "Tarih Islemleri";
154:         //
155:         // btnTarihAraligi
156:         //
157:         this.btnTarihAraligi.Location = new System.Drawing.Point(204, 76);
158:         this.btnTarihAraligi.Name = "btnTarihAraligi";
159:         this.btnTarihAraligi.Size = new System.Drawing.Size(76, 23);
160:         this.btnTarihAraligi.TabIndex = 10;
161:         this.btnTarihAraligi.Text = "Tarih Araligi";
162:         this.btnTarihAraligi.UseVisualStyleBackColor = true;
163:         this.btnTarihAraligi.Click += new
System.EventHandler(this.btnTarihAraligi_Click);
164:         //
165:         // btnYasHesaplaMetinsel
166:         //
167:         this.btnYasHesaplaMetinsel.Location = new System.Drawing.Point(84, 76);
168:         this.btnYasHesaplaMetinsel.Name = "btnYasHesaplaMetinsel";
169:         this.btnYasHesaplaMetinsel.Size = new System.Drawing.Size(113, 23);
170:         this.btnYasHesaplaMetinsel.TabIndex = 9;
171:         this.btnYasHesaplaMetinsel.Text = "Yas Hesapla (Metin)";
172:         this.btnYasHesaplaMetinsel.UseVisualStyleBackColor = true;
173:         this.btnYasHesaplaMetinsel.Click += new
System.EventHandler(this.btnYasHesaplaMetinsel_Click);
174:         //
175:         // btnYasHesapla
176:         //
177:         this.btnYasHesapla.Location = new System.Drawing.Point(3, 76);
178:         this.btnYasHesapla.Name = "btnYasHesapla";
179:         this.btnYasHesapla.Size = new System.Drawing.Size(75, 23);
180:         this.btnYasHesapla.TabIndex = 8;
181:         this.btnYasHesapla.Text = "Yas Hesapla";
182:         this.btnYasHesapla.UseVisualStyleBackColor = true;
183:         this.btnYasHesapla.Click += new
System.EventHandler(this.btnYasHesapla_Click);
184:         //
185:         // label4
186:         //
187:         this.label4.AutoSize = true;
188:         this.label4.Location = new System.Drawing.Point(51, 109);
189:         this.label4.Name = "label4";
190:         this.label4.Size = new System.Drawing.Size(38, 13);
191:         this.label4.TabIndex = 7;

```

```

192:         this.label4.Text = "Sonuç";
193:         //
194:         // lbSonuc
195:         //
196:         this.lbSonuc.FormattingEnabled = true;
197:         this.lbSonuc.HorizontalScrollbar = true;
198:         this.lbSonuc.Location = new System.Drawing.Point(117, 109);
199:         this.lbSonuc.Name = "lbSonuc";
200:         this.lbSonuc.Size = new System.Drawing.Size(200, 69);
201:         this.lbSonuc.TabIndex = 6;
202:         //
203:         // dtpBitisTarihi
204:         //
205:         this.dtpBitisTarihi.Format =
System.Windows.Forms.DateTimePickerFormat.Short;
206:         this.dtpBitisTarihi.Location = new System.Drawing.Point(117, 43);
207:         this.dtpBitisTarihi.Name = "dtpBitisTarihi";
208:         this.dtpBitisTarihi.ShowCheckBox = true;
209:         this.dtpBitisTarihi.Size = new System.Drawing.Size(200, 20);
210:         this.dtpBitisTarihi.TabIndex = 5;
211:         //
212:         // dtpBaslangicTarihi
213:         //
214:         this.dtpBaslangicTarihi.Format =
System.Windows.Forms.DateTimePickerFormat.Short;
215:         this.dtpBaslangicTarihi.Location = new System.Drawing.Point(117, 17);
216:         this.dtpBaslangicTarihi.Name = "dtpBaslangicTarihi";
217:         this.dtpBaslangicTarihi.ShowCheckBox = true;
218:         this.dtpBaslangicTarihi.Size = new System.Drawing.Size(200, 20);
219:         this.dtpBaslangicTarihi.TabIndex = 4;
220:         //
221:         // label3
222:         //
223:         this.label3.AutoSize = true;
224:         this.label3.Location = new System.Drawing.Point(44, 47);
225:         this.label3.Name = "label3";
226:         this.label3.Size = new System.Drawing.Size(55, 13);
227:         this.label3.TabIndex = 3;
228:         this.label3.Text = "Bitis Tarihi";
229:         //
230:         // label2
231:         //
232:         this.label2.AutoSize = true;
233:         this.label2.Location = new System.Drawing.Point(17, 24);
234:         this.label2.Name = "label2";
235:         this.label2.Size = new System.Drawing.Size(82, 13);
236:         this.label2.TabIndex = 2;
237:         this.label2.Text = "Baslangic Tarihi";
238:         //
239:         // pnlSifreIslemleri
240:         //
241:         this.pnlSifreIslemleri.BorderStyle =
System.Windows.Forms.BorderStyle.FixedSingle;
242:         this.pnlSifreIslemleri.Controls.Add(this.label7);
243:         this.pnlSifreIslemleri.Controls.Add(this.btnSifreUygunMu);
244:         this.pnlSifreIslemleri.Controls.Add(this.btnRastgeleSifreUret);
245:         this.pnlSifreIslemleri.Controls.Add(this.btnSha512Hash);
246:         this.pnlSifreIslemleri.Controls.Add(this.btnSha256Hash);
247:         this.pnlSifreIslemleri.Controls.Add(this.label6);
248:         this.pnlSifreIslemleri.Controls.Add(this.lbSifreSonuc);
249:         this.pnlSifreIslemleri.Controls.Add(this.btnMd5Hash);
250:         this.pnlSifreIslemleri.Controls.Add(this.txtSifrelenecekMetin);
251:         this.pnlSifreIslemleri.Controls.Add(this.label5);
252:         this.pnlSifreIslemleri.Location = new System.Drawing.Point(411, 56);
253:         this.pnlSifreIslemleri.Name = "pnlSifreIslemleri";
254:         this.pnlSifreIslemleri.Size = new System.Drawing.Size(377, 181);
255:         this.pnlSifreIslemleri.TabIndex = 3;
256:         //
257:         // label7

```

```

258:         //
259:         this.label7.AutoSize = true;
260:         this.label7.Dock = System.Windows.Forms.DockStyle.Top;
261:         this.label7.Location = new System.Drawing.Point(0, 0);
262:         this.label7.Name = "label7";
263:         this.label7.Size = new System.Drawing.Size(68, 13);
264:         this.label7.TabIndex = 18;
265:         this.label7.Text = "Sifre Islemleri";
266:         this.label7.TextAlign = System.Drawing.ContentAlignment.TopCenter;
267:         //
268:         // btnSifreUygunMu
269:         //
270:         this.btnSifreUygunMu.Location = new System.Drawing.Point(173, 71);
271:         this.btnSifreUygunMu.Name = "btnSifreUygunMu";
272:         this.btnSifreUygunMu.Size = new System.Drawing.Size(145, 23);
273:         this.btnSifreUygunMu.TabIndex = 17;
274:         this.btnSifreUygunMu.Text = "Sifre Uygun Mu?";
275:         this.btnSifreUygunMu.UseVisualStyleBackColor = true;
276:         this.btnSifreUygunMu.Click += new
System.EventHandler(this.btnSifreUygunMu_Click);
277:         //
278:         // btnRastgeleSifreUret
279:         //
280:         this.btnRastgeleSifreUret.Location = new System.Drawing.Point(21, 71);
281:         this.btnRastgeleSifreUret.Name = "btnRastgeleSifreUret";
282:         this.btnRastgeleSifreUret.Size = new System.Drawing.Size(145, 23);
283:         this.btnRastgeleSifreUret.TabIndex = 16;
284:         this.btnRastgeleSifreUret.Text = "Rastgele Sifre Üret";
285:         this.btnRastgeleSifreUret.UseVisualStyleBackColor = true;
286:         this.btnRastgeleSifreUret.Click += new
System.EventHandler(this.btnRastgeleSifreUret_Click);
287:         //
288:         // btnSha512Hash
289:         //
290:         this.btnSha512Hash.Location = new System.Drawing.Point(243, 42);
291:         this.btnSha512Hash.Name = "btnSha512Hash";
292:         this.btnSha512Hash.Size = new System.Drawing.Size(75, 23);
293:         this.btnSha512Hash.TabIndex = 15;
294:         this.btnSha512Hash.Text = "SHA 512";
295:         this.btnSha512Hash.UseVisualStyleBackColor = true;
296:         this.btnSha512Hash.Click += new
System.EventHandler(this.btnSha512Hash_Click);
297:         //
298:         // btnSha256Hash
299:         //
300:         this.btnSha256Hash.Location = new System.Drawing.Point(131, 42);
301:         this.btnSha256Hash.Name = "btnSha256Hash";
302:         this.btnSha256Hash.Size = new System.Drawing.Size(75, 23);
303:         this.btnSha256Hash.TabIndex = 14;
304:         this.btnSha256Hash.Text = "SHA 256";
305:         this.btnSha256Hash.UseVisualStyleBackColor = true;
306:         this.btnSha256Hash.Click += new
System.EventHandler(this.btnSha256Hash_Click);
307:         //
308:         // label6
309:         //
310:         this.label6.AutoSize = true;
311:         this.label6.Location = new System.Drawing.Point(74, 104);
312:         this.label6.Name = "label6";
313:         this.label6.Size = new System.Drawing.Size(38, 13);
314:         this.label6.TabIndex = 13;
315:         this.label6.Text = "Sonuç";
316:         //
317:         // lbSifreSonuc
318:         //
319:         this.lbSifreSonuc.FormattingEnabled = true;
320:         this.lbSifreSonuc.HorizontalScrollbar = true;
321:         this.lbSifreSonuc.Location = new System.Drawing.Point(118, 100);
322:         this.lbSifreSonuc.Name = "lbSifreSonuc";

```

```

323:         this.lbSifreSonuc.Size = new System.Drawing.Size(200, 69);
324:         this.lbSifreSonuc.TabIndex = 12;
325:         //
326:         // btnMd5Hash
327:         //
328:         this.btnMd5Hash.Location = new System.Drawing.Point(21, 42);
329:         this.btnMd5Hash.Name = "btnMd5Hash";
330:         this.btnMd5Hash.Size = new System.Drawing.Size(75, 23);
331:         this.btnMd5Hash.TabIndex = 11;
332:         this.btnMd5Hash.Text = "MD5";
333:         this.btnMd5Hash.UseVisualStyleBackColor = true;
334:         this.btnMd5Hash.Click += new System.EventHandler(this.btnMd5Hash_Click);
335:         //
336:         // txtSifrelenecekMetin
337:         //
338:         this.txtSifrelenecekMetin.Location = new System.Drawing.Point(118, 16);
339:         this.txtSifrelenecekMetin.Name = "txtSifrelenecekMetin";
340:         this.txtSifrelenecekMetin.Size = new System.Drawing.Size(199, 20);
341:         this.txtSifrelenecekMetin.TabIndex = 4;
342:         //
343:         // label5
344:         //
345:         this.label5.AutoSize = true;
346:         this.label5.Location = new System.Drawing.Point(17, 19);
347:         this.label5.Name = "label5";
348:         this.label5.Size = new System.Drawing.Size(95, 13);
349:         this.label5.TabIndex = 3;
350:         this.label5.Text = "Sifrelenecek Metin";
351:         //
352:         // pnlSayiIslemleri
353:         //
354:         this.pnlSayiIslemleri.BorderStyle =
System.Windows.Forms.BorderStyle.FixedSingle;
355:         this.pnlSayiIslemleri.Controls.Add(this.label9);
356:         this.pnlSayiIslemleri.Controls.Add(this.btnOnayKoduOlustur);
357:         this.pnlSayiIslemleri.Controls.Add(this.btnMetneDonustur);
358:         this.pnlSayiIslemleri.Controls.Add(this.label10);
359:         this.pnlSayiIslemleri.Controls.Add(this.lbSayiSonuc);
360:         this.pnlSayiIslemleri.Controls.Add(this.btnSayisalMi);
361:         this.pnlSayiIslemleri.Controls.Add(this.txtSayi);
362:         this.pnlSayiIslemleri.Controls.Add(this.label11);
363:         this.pnlSayiIslemleri.Location = new System.Drawing.Point(15, 257);
364:         this.pnlSayiIslemleri.Name = "pnlSayiIslemleri";
365:         this.pnlSayiIslemleri.Size = new System.Drawing.Size(377, 181);
366:         this.pnlSayiIslemleri.TabIndex = 19;
367:         //
368:         // label9
369:         //
370:         this.label9.AutoSize = true;
371:         this.label9.Dock = System.Windows.Forms.DockStyle.Top;
372:         this.label9.Location = new System.Drawing.Point(0, 0);
373:         this.label9.Name = "label9";
374:         this.label9.Size = new System.Drawing.Size(67, 13);
375:         this.label9.TabIndex = 18;
376:         this.label9.Text = "Sayi Islemleri";
377:         this.label9.TextAlign = System.Drawing.ContentAlignment.TopCenter;
378:         //
379:         // btnOnayKoduOlustur
380:         //
381:         this.btnOnayKoduOlustur.Location = new System.Drawing.Point(243, 42);
382:         this.btnOnayKoduOlustur.Name = "btnOnayKoduOlustur";
383:         this.btnOnayKoduOlustur.Size = new System.Drawing.Size(75, 23);
384:         this.btnOnayKoduOlustur.TabIndex = 15;
385:         this.btnOnayKoduOlustur.Text = "Onay Kodu";
386:         this.btnOnayKoduOlustur.UseVisualStyleBackColor = true;
387:         this.btnOnayKoduOlustur.Click += new
System.EventHandler(this.btnOnayKoduOlustur_Click);
388:         //
389:         // btnMetneDonustur

```



```

390:         //
391:         this.btnMetneDonustur.Location = new System.Drawing.Point(102, 42);
392:         this.btnMetneDonustur.Name = "btnMetneDonustur";
393:         this.btnMetneDonustur.Size = new System.Drawing.Size(135, 23);
394:         this.btnMetneDonustur.TabIndex = 14;
395:         this.btnMetneDonustur.Text = "Para Metnine Dönüştür";
396:         this.btnMetneDonustur.UseVisualStyleBackColor = true;
397:         this.btnMetneDonustur.Click += new
System.EventHandler(this.btnMetneDonustur_Click);
398:         //
399:         // label10
400:         //
401:         this.label10.AutoSize = true;
402:         this.label10.Location = new System.Drawing.Point(18, 78);
403:         this.label10.Name = "label10";
404:         this.label10.Size = new System.Drawing.Size(38, 13);
405:         this.label10.TabIndex = 13;
406:         this.label10.Text = "Sonuç";
407:         //
408:         // lbSayiSonuc
409:         //
410:         this.lbSayiSonuc.FormattingEnabled = true;
411:         this.lbSayiSonuc.HorizontalScrollbar = true;
412:         this.lbSayiSonuc.Location = new System.Drawing.Point(62, 78);
413:         this.lbSayiSonuc.Name = "lbSayiSonuc";
414:         this.lbSayiSonuc.Size = new System.Drawing.Size(254, 82);
415:         this.lbSayiSonuc.TabIndex = 12;
416:         //
417:         // btnSayisalMi
418:         //
419:         this.btnSayisalMi.Location = new System.Drawing.Point(21, 42);
420:         this.btnSayisalMi.Name = "btnSayisalMi";
421:         this.btnSayisalMi.Size = new System.Drawing.Size(75, 23);
422:         this.btnSayisalMi.TabIndex = 11;
423:         this.btnSayisalMi.Text = "Sayisal Mi?";
424:         this.btnSayisalMi.UseVisualStyleBackColor = true;
425:         this.btnSayisalMi.Click += new
System.EventHandler(this.btnSayisalMi_Click);
426:         //
427:         // txtSayi
428:         //
429:         this.txtSayi.Location = new System.Drawing.Point(118, 16);
430:         this.txtSayi.Name = "txtSayi";
431:         this.txtSayi.Size = new System.Drawing.Size(199, 20);
432:         this.txtSayi.TabIndex = 4;
433:         //
434:         // label11
435:         //
436:         this.label11.AutoSize = true;
437:         this.label11.Location = new System.Drawing.Point(17, 19);
438:         this.label11.Name = "label11";
439:         this.label11.Size = new System.Drawing.Size(95, 13);
440:         this.label11.TabIndex = 3;
441:         this.label11.Text = "Sifrelenecek Metin";
442:         //
443:         // pnlKisiIslemleri
444:         //
445:         this.pnlKisiIslemleri.BorderStyle =
System.Windows.Forms.BorderStyle.FixedSingle;
446:         this.pnlKisiIslemleri.Controls.Add(this.label12);
447:         this.pnlKisiIslemleri.Controls.Add(this.btnEPostaDogrula);
448:         this.pnlKisiIslemleri.Controls.Add(this.btnIbanDogrula);
449:         this.pnlKisiIslemleri.Controls.Add(this.label13);
450:         this.pnlKisiIslemleri.Controls.Add(this.lbKisiSonuc);
451:         this.pnlKisiIslemleri.Controls.Add(this.btnTcKimlikDogrula);
452:         this.pnlKisiIslemleri.Controls.Add(this.txtKisiMetni);
453:         this.pnlKisiIslemleri.Controls.Add(this.label14);
454:         this.pnlKisiIslemleri.Location = new System.Drawing.Point(411, 257);
455:         this.pnlKisiIslemleri.Name = "pnlKisiIslemleri";

```

```

456:         this.pnlKisiIslemleri.Size = new System.Drawing.Size(377, 181);
457:         this.pnlKisiIslemleri.TabIndex = 20;
458:         //
459:         // label12
460:         //
461:         this.label12.AutoSize = true;
462:         this.label12.Dock = System.Windows.Forms.DockStyle.Top;
463:         this.label12.Location = new System.Drawing.Point(0, 0);
464:         this.label12.Name = "label12";
465:         this.label12.Size = new System.Drawing.Size(63, 13);
466:         this.label12.TabIndex = 18;
467:         this.label12.Text = "Kisi Islemleri";
468:         this.label12.TextAlign = System.Drawing.ContentAlignment.TopCenter;
469:         //
470:         // btnEPostaDogrula
471:         //
472:         this.btnEPostaDogrula.Location = new System.Drawing.Point(223, 42);
473:         this.btnEPostaDogrula.Name = "btnEPostaDogrula";
474:         this.btnEPostaDogrula.Size = new System.Drawing.Size(95, 23);
475:         this.btnEPostaDogrula.TabIndex = 15;
476:         this.btnEPostaDogrula.Text = "E-Posta Dogrula";
477:         this.btnEPostaDogrula.UseVisualStyleBackColor = true;
478:         this.btnEPostaDogrula.Click += new
System.EventHandler(this.btnEPostaDogrula_Click);
479:         //
480:         // btnIbanDogrula
481:         //
482:         this.btnIbanDogrula.Location = new System.Drawing.Point(131, 42);
483:         this.btnIbanDogrula.Name = "btnIbanDogrula";
484:         this.btnIbanDogrula.Size = new System.Drawing.Size(86, 23);
485:         this.btnIbanDogrula.TabIndex = 14;
486:         this.btnIbanDogrula.Text = "IBAN Dogrula";
487:         this.btnIbanDogrula.UseVisualStyleBackColor = true;
488:         this.btnIbanDogrula.Click += new
System.EventHandler(this.btnIbanDogrula_Click);
489:         //
490:         // label13
491:         //
492:         this.label13.AutoSize = true;
493:         this.label13.Location = new System.Drawing.Point(18, 78);
494:         this.label13.Name = "label13";
495:         this.label13.Size = new System.Drawing.Size(38, 13);
496:         this.label13.TabIndex = 13;
497:         this.label13.Text = "Sonuç";
498:         //
499:         // lbKisiSonuc
500:         //
501:         this.lbKisiSonuc.FormattingEnabled = true;
502:         this.lbKisiSonuc.HorizontalScrollbar = true;
503:         this.lbKisiSonuc.Location = new System.Drawing.Point(62, 78);
504:         this.lbKisiSonuc.Name = "lbKisiSonuc";
505:         this.lbKisiSonuc.Size = new System.Drawing.Size(254, 82);
506:         this.lbKisiSonuc.TabIndex = 12;
507:         //
508:         // btnTcKimlikDogrula
509:         //
510:         this.btnTcKimlikDogrula.Location = new System.Drawing.Point(21, 42);
511:         this.btnTcKimlikDogrula.Name = "btnTcKimlikDogrula";
512:         this.btnTcKimlikDogrula.Size = new System.Drawing.Size(104, 23);
513:         this.btnTcKimlikDogrula.TabIndex = 11;
514:         this.btnTcKimlikDogrula.Text = "TC Kimlik Dogrula";
515:         this.btnTcKimlikDogrula.UseVisualStyleBackColor = true;
516:         this.btnTcKimlikDogrula.Click += new
System.EventHandler(this.btnTcKimlikDogrula_Click);
517:         //
518:         // txtKisiMetni
519:         //
520:         this.txtKisiMetni.Location = new System.Drawing.Point(118, 16);
521:         this.txtKisiMetni.Name = "txtKisiMetni";

```

```

522:         this.txtKisiMetni.Size = new System.Drawing.Size(199, 20);
523:         this.txtKisiMetni.TabIndex = 4;
524:         //
525:         // label14
526:         //
527:         this.label14.AutoSize = true;
528:         this.label14.Location = new System.Drawing.Point(17, 19);
529:         this.label14.Name = "label14";
530:         this.label14.Size = new System.Drawing.Size(33, 13);
531:         this.label14.TabIndex = 3;
532:         this.label14.Text = "Metin";
533:         //
534:         // pnlMetinIslemleri
535:         //
536:         this.pnlMetinIslemleri.BorderStyle =
System.Windows.Forms.BorderStyle.FixedSingle;
537:         this.pnlMetinIslemleri.Controls.Add(this.btnTelefonNoDuzenle);
538:         this.pnlMetinIslemleri.Controls.Add(this.btnMetinSifrele);
539:         this.pnlMetinIslemleri.Controls.Add(this.label15);
540:         this.pnlMetinIslemleri.Controls.Add(this.btnTurkceKarakterDuzelt);
541:         this.pnlMetinIslemleri.Controls.Add(this.btnIlkHarfiBuyukYap);
542:         this.pnlMetinIslemleri.Controls.Add(this.label16);
543:         this.pnlMetinIslemleri.Controls.Add(this.lbMetinSonuc);
544:         this.pnlMetinIslemleri.Controls.Add(this.btnIlkHarfleriBuyukYap);
545:         this.pnlMetinIslemleri.Controls.Add(this.txtMetin);
546:         this.pnlMetinIslemleri.Controls.Add(this.label17);
547:         this.pnlMetinIslemleri.Location = new System.Drawing.Point(15, 458);
548:         this.pnlMetinIslemleri.Name = "pnlMetinIslemleri";
549:         this.pnlMetinIslemleri.Size = new System.Drawing.Size(773, 181);
550:         this.pnlMetinIslemleri.TabIndex = 21;
551:         //
552:         // btnTelefonNoDuzenle
553:         //
554:         this.btnTelefonNoDuzenle.Location = new System.Drawing.Point(503, 42);
555:         this.btnTelefonNoDuzenle.Name = "btnTelefonNoDuzenle";
556:         this.btnTelefonNoDuzenle.Size = new System.Drawing.Size(134, 23);
557:         this.btnTelefonNoDuzenle.TabIndex = 20;
558:         this.btnTelefonNoDuzenle.Text = "Telefon No Düzenle";
559:         this.btnTelefonNoDuzenle.UseVisualStyleBackColor = true;
560:         this.btnTelefonNoDuzenle.Click += new
System.EventHandler(this.btnTelefonNoDuzenle_Click);
561:         //
562:         // btnMetinSifrele
563:         //
564:         this.btnMetinSifrele.Location = new System.Drawing.Point(363, 42);
565:         this.btnMetinSifrele.Name = "btnMetinSifrele";
566:         this.btnMetinSifrele.Size = new System.Drawing.Size(134, 23);
567:         this.btnMetinSifrele.TabIndex = 19;
568:         this.btnMetinSifrele.Text = "Metin Sifrele";
569:         this.btnMetinSifrele.UseVisualStyleBackColor = true;
570:         this.btnMetinSifrele.Click += new
System.EventHandler(this.btnMetinSifrele_Click);
571:         //
572:         // label15
573:         //
574:         this.label15.AutoSize = true;
575:         this.label15.Dock = System.Windows.Forms.DockStyle.Top;
576:         this.label15.Location = new System.Drawing.Point(0, 0);
577:         this.label15.Name = "label15";
578:         this.label15.Size = new System.Drawing.Size(73, 13);
579:         this.label15.TabIndex = 18;
580:         this.label15.Text = "Metin Islemleri";
581:         this.label15.TextAlign = System.Drawing.ContentAlignment.TopCenter;
582:         //
583:         // btnTurkceKarakterDuzelt
584:         //
585:         this.btnTurkceKarakterDuzelt.Location = new System.Drawing.Point(223, 42);
586:         this.btnTurkceKarakterDuzelt.Name = "btnTurkceKarakterDuzelt";
587:         this.btnTurkceKarakterDuzelt.Size = new System.Drawing.Size(134, 23);

```



```

588:         this.btnTürkçeKarakterDüzeltil.TabIndex = 15;
589:         this.btnTürkçeKarakterDüzeltil.Text = "Türkçe Karakter Düzelt";
590:         this.btnTürkçeKarakterDüzeltil.UseVisualStyleBackColor = true;
591:         this.btnTürkçeKarakterDüzeltil.Click += new
System.EventHandler(this.btnTürkçeKarakterDüzeltil_Click);
592:         //
593:         // btnİlkHarfiBuyukYap
594:         //
595:         this.btnİlkHarfiBuyukYap.Location = new System.Drawing.Point(131, 42);
596:         this.btnİlkHarfiBuyukYap.Name = "btnİlkHarfiBuyukYap";
597:         this.btnİlkHarfiBuyukYap.Size = new System.Drawing.Size(86, 23);
598:         this.btnİlkHarfiBuyukYap.TabIndex = 14;
599:         this.btnİlkHarfiBuyukYap.Text = "İlk Harf Büyük";
600:         this.btnİlkHarfiBuyukYap.UseVisualStyleBackColor = true;
601:         this.btnİlkHarfiBuyukYap.Click += new
System.EventHandler(this.btnİlkHarfiBuyukYap_Click);
602:         //
603:         // label16
604:         //
605:         this.label16.AutoSize = true;
606:         this.label16.Location = new System.Drawing.Point(18, 78);
607:         this.label16.Name = "label16";
608:         this.label16.Size = new System.Drawing.Size(38, 13);
609:         this.label16.TabIndex = 13;
610:         this.label16.Text = "Sonuç";
611:         //
612:         // lbMetinSonuc
613:         //
614:         this.lbMetinSonuc.FormattingEnabled = true;
615:         this.lbMetinSonuc.HorizontalScrollbar = true;
616:         this.lbMetinSonuc.Location = new System.Drawing.Point(62, 78);
617:         this.lbMetinSonuc.Name = "lbMetinSonuc";
618:         this.lbMetinSonuc.Size = new System.Drawing.Size(652, 82);
619:         this.lbMetinSonuc.TabIndex = 12;
620:         //
621:         // btnİlkHarfleriBuyukYap
622:         //
623:         this.btnİlkHarfleriBuyukYap.Location = new System.Drawing.Point(21, 42);
624:         this.btnİlkHarfleriBuyukYap.Name = "btnİlkHarfleriBuyukYap";
625:         this.btnİlkHarfleriBuyukYap.Size = new System.Drawing.Size(104, 23);
626:         this.btnİlkHarfleriBuyukYap.TabIndex = 11;
627:         this.btnİlkHarfleriBuyukYap.Text = "İlk Harfler Büyük";
628:         this.btnİlkHarfleriBuyukYap.UseVisualStyleBackColor = true;
629:         this.btnİlkHarfleriBuyukYap.Click += new
System.EventHandler(this.btnİlkHarfleriBuyukYap_Click);
630:         //
631:         // txtMetin
632:         //
633:         this.txtMetin.Location = new System.Drawing.Point(119, 12);
634:         this.txtMetin.Name = "txtMetin";
635:         this.txtMetin.Size = new System.Drawing.Size(595, 20);
636:         this.txtMetin.TabIndex = 4;
637:         //
638:         // label17
639:         //
640:         this.label17.AutoSize = true;
641:         this.label17.Location = new System.Drawing.Point(17, 19);
642:         this.label17.Name = "label17";
643:         this.label17.Size = new System.Drawing.Size(33, 13);
644:         this.label17.TabIndex = 3;
645:         this.label17.Text = "Metin";
646:         //
647:         // FrmAnaSayfa
648:         //
649:         this.AutoScaleDimensions = new System.Drawing.SizeF(6F, 13F);
650:         this.AutoScaleMode = System.Windows.Forms.AutoScaleMode.Font;
651:         this.AutoSizeMode = System.Windows.Forms.AutoSizeMode.GrowAndShrink;
652:         this.ClientSize = new System.Drawing.Size(800, 686);
653:         this.Controls.Add(this.pnlMetinIslemleri);

```

```

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

```

```

723:         private System.Windows.Forms.Label label14;
724:         private System.Windows.Forms.Panel pnlMetinIslemleri;
725:         private System.Windows.Forms.Label label15;
726:         private System.Windows.Forms.Button btnTurkceKarakterDuzelt;
727:         private System.Windows.Forms.Button btnIlkHarfiBuyukYap;
728:         private System.Windows.Forms.Label label16;
729:         private System.Windows.Forms.ListBox lbMetinSonuc;
730:         private System.Windows.Forms.Button btnIlkHarfleriBuyukYap;
731:         private System.Windows.Forms.TextBox txtMetin;
732:         private System.Windows.Forms.Label label17;
733:         private System.Windows.Forms.Button btnMetinSifrele;
734:         private System.Windows.Forms.Button btnTelefonNoDuzenle;
735:     }
736: }
737:

```

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 metotlarin tamamini içeren sinif
9:     /// </summary>
10:    public static class KisiIslemleri
11:    {
12:        /// <summary>
13:        /// Girilen TC Kimlik Numarasinin, standart TC Kimlik No algoritmasına uygun
14:        /// olmadigini kontrol eden metottur.
15:        /// </summary>
16:        /// <param name="tcKimlikNo">TC Kimlik No</param>
17:        /// <returns>TC Kimlik No'nun Uygun Olup Olmadigi Bilgisi</returns>
18:        public static bool DogrulaTcKimlikNo(string tcKimlikNo)
19:        {
20:            var tcNo = Convert.ToInt64(tcKimlikNo);
21:            var atcno = tcNo / 100;
22:            var btcno = tcNo / 100;
23:            var c1 = atcno % 10; atcno /= 10;
24:            var c2 = atcno % 10; atcno /= 10;
25:            var c3 = atcno % 10; atcno /= 10;
26:            var c4 = atcno % 10; atcno /= 10;
27:            var c5 = atcno % 10; atcno /= 10;
28:            var c6 = atcno % 10; atcno /= 10;
29:            var c7 = atcno % 10; atcno /= 10;
30:            var c8 = atcno % 10; atcno /= 10;
31:            var c9 = atcno % 10;
32:            var q1 = (10 - ((c1 + c3 + c5 + c7 + c9) * 3 + c2 + c4 + c6 + c8) % 10) %
10;
33:            var q2 = (10 - ((c2 + c4 + c6 + c8 + q1) * 3 + c1 + c3 + c5 + c7 + c9) %
10) % 10;
34:            var dogrula = btcno * 100 + q1 * 10 + q2 == tcNo;
35:            return dogrula;

```

```

36:     }
37:
38:     /// <summary>
39:     /// Girilen IBAN'in standar IBAN algoritmasına 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-]+\.[\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 yapılması için hazırlanan metottur
151:     /// </summary>
152:     /// <param name="input">Girilen metin</param>
153:     /// <returns>Metnin Kelimelerinin İlk Harflerinin Büyüültülmüş 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:     /// <summary>
186:     /// Girilen metnin bosluklarinin kaldırilmasini saglayan metottur.
187:     /// </summary>
188:     /// <param name="metin">Telefon No Bilgisi</param>
189:     /// <returns>Girilen Metnin Istenen Hali</returns>
190:

```



```

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:

```

```

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 SayiHalineCevir(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, CultureInfo.InvariantCulture, out val))
34:                    return 0;
35:                return val / 100;
36:            }
37:            var value = decimal.TryParse(metin.Replace(",", "").Replace(".", "",
38:            " ").Replace("_", " "), NumberStyles.Number, CultureInfo.InvariantCulture, out val);
39:            return value ? val : 0;
40:        }
41:        /// <summary>
42:        /// Elimizdeki tutar verilerinin
43:        /// Yazı formatında gösterilmesini sağlayacak bir metottur
44:        /// </summary>

```

```

45:         /// <param name="tutar">Girilen Sayi Bilgisi</param>
46:         /// <returns>Sayinin Yazıya Dönüştürülmüş Hali</returns>
47:         public static string SayıYazıyaCevir(decimal tutar)
48:         {
49:             var sTutar = tutar.ToString("F2").Replace('.', ','); // Replace('.', ',')
ondalık ayracının . olma durumu için
50:             var lira = sTutar.Substring(0, sTutar.IndexOf(',')); //tutarın tam kısmı
51:             var kurus = sTutar.Substring(sTutar.IndexOf(',') + 1, 2);
52:             var yazi = "";
53:
54:             var birler = new[] { "", "BİR", "İKİ", "ÜÇ", "DÖRT", "BES", "ALTI",
"YEDİ", "SEKİZ", "DOKUZ" };
55:             var onlar = new[] { "", "ON", "YIRMI", "OTUZ", "KIRK", "ELLİ", "ALTMIS",
"YETMİS", "SEKSEN", "DOKSAN" };
56:             var binler = new[] { "KATRİLYON", "TRİLYON", "MİLYAR", "MİLYON", "BIN", ""
};
57:
58:             const int grupSayisi = 6;
59:             //sayıdaki 3'lü grup sayısı. katrilyon için 6. (1.234,00 daki grup sayısı
2'dir.)
60:             //KATRİLYON'un basına ekleyeceğiniz her değer için grup sayisini artiriniz.
61:
62:             lira = lira.PadLeft(grupSayisi * 3, '0'); //sayinin soluna '0' eklenerek
sayi 'grup sayısı x 3' basakmakli yapıyor.
63:
64:             for (var i = 0; i < grupSayisi * 3; i += 3) //sayi 3'erli gruplar halinde
ele aliniyor.
65:             {
66:                 var grupDegeri = "";
67:
68:                 if (lira.Substring(i, 1) != "0")
69:                     grupDegeri += birler[Convert.ToInt32(lira.Substring(i, 1))] +
"YÜZ"; //yüzler
70:
71:                 if (grupDegeri == "BİRYÜZ") //biryüz düzeltiliyor.
72:                     grupDegeri = "YÜZ";
73:
74:                 grupDegeri += onlar[Convert.ToInt32(lira.Substring(i + 1, 1))]; //onlar
75:
76:                 grupDegeri += birler[Convert.ToInt32(lira.Substring(i + 2, 1))];
//birler
77:
78:                 if (grupDegeri != "") //binler
79:                     grupDegeri += binler[i / 3];
80:
81:                 if (grupDegeri == "BİRBİN") //birbin düzeltiliyor.
82:                     grupDegeri = "BİN";
83:
84:                 yazi += grupDegeri;
85:             }
86:
87:             if (yazi != "")
88:                 yazi += " TL ";
89:
90:             var yaziUzunlugu = yazi.Length;
91:
92:             if (kurus.Substring(0, 1) != "0") //kurus onlar
93:                 yazi += onlar[Convert.ToInt32(kurus.Substring(0, 1))];
94:
95:             if (kurus.Substring(1, 1) != "0") //kurus birler
96:                 yazi += birler[Convert.ToInt32(kurus.Substring(1, 1))];
97:
98:             if (yazi.Length > yaziUzunlugu)
99:                 yazi += " Kr.";
100:             else
101:                 yazi += "SİFİR Kr.";
102:
103:             return yazi;
104:         }

```

```

105:
106:     /// <summary>
107:     /// Sadece rakamlardan olusan 6 haneli random sayi üreten metottur.
108:     /// </summary>
109:     /// <returns>6 Haneli Onay Kodu</returns>
110:     public static string AltihaneliOnayKoduOlustur()
111:     {
112:         var rastgele = new Random();
113:         var sb = new StringBuilder();
114:         for (var i = 0; i < 6; i++)
115:         {
116:             var ascii = rastgele.Next(48, 57); //Rakamlar
117:             var karakter = Convert.ToChar(ascii);
118:             sb.Append(karakter);
119:         }
120:         return sb.ToString();
121:     }
122: }
123: }

```

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: }

```

```

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

```

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) => current +
25:             $" {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 = SHA256Managed.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:

```

```

49:         /// <param name="text">Sifrelenecek Metin</param>
50:         /// <returns>Girdi Metninin Sifrelenmis Hali</returns>
51:         public static string HesaplaMD5(string text)
52:         {
53:             var md5 = MD5.Create();
54:             var inputBytes = Encoding.ASCII.GetBytes(text);
55:             var hash = md5.ComputeHash(inputBytes);
56:
57:             var sb = new StringBuilder();
58:             foreach (var t in hash)
59:             {
60:                 sb.Append(t.ToString("X2"));
61:             }
62:             return sb.ToString();
63:         }
64:
65:         /// <summary>
66:         /// sifreyi kriterlere göre kontrol edip
67:         /// uymayan kosullarini string listesine geri 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></returns>
103:         public static string RastgeleSifreUret()
104:         {
105:             var sifreUzunluk = 8;
106:             const string gecerliKarakterler =
"abcdefghijklmnopqrstuvwxyzABCDEFGHIJKLMNOPQRSTUVWXYZ1234567890";
107:             var strB = new StringBuilder(100);
108:             var random = new Random();
109:             while (0 < sifreUzunluk--)
110:             {
111:                 strB.Append(gecerliKarakterler[random.Next(gecerliKarakterler.Length)]);
112:             }

```

```

113:         return strB.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:
5: namespace CSharpHelperMethods.YardimciSiniflar
6: {
7:     /// <summary>
8:     /// Alfa numerik siralama için hazirlanmis bir metod
9:     /// </summary>
10:    public static class SiralamaIslemleri
11:    {
12:        private enum ChunkType { Alphanumeric, Numeric }
13:
14:        private static bool InChunk(char ch, char otherCh)
15:        {
16:            var type = ChunkType.Alphanumeric;
17:
18:            if (char.IsDigit(otherCh))
19:            {
20:                type = ChunkType.Numeric;
21:            }
22:
23:            return (type != ChunkType.Alphanumeric || !char.IsDigit(ch)) && (type !=
ChunkType.Numeric || char.IsDigit(ch));
24:        }
25:
26:        public static int Compare(string x, string y)
27:        {
28:            if (string.IsNullOrEmpty(x) || string.IsNullOrEmpty(y)) return 0;
29:            var thisMarker = 0;
30:            var thatMarker = 0;
31:
32:            while (thisMarker < x.Length || thatMarker < y.Length)
33:            {
34:                if (thisMarker >= x.Length)
35:                {
36:                    return -1;
37:                }
38:                if (thatMarker >= y.Length)
39:                {
40:                    return 1;
41:                }
42:
43:                var thisCh = x[thisMarker];
44:                var thatCh = y[thatMarker];
45:
46:                var thisChunk = new StringBuilder();
47:                var thatChunk = new StringBuilder();
48:

```

```

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

```

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:
3: namespace CSharpHelperMethods.YardimciSiniflar
4: {
5:     /// <summary>
6:     /// Tarihsel veri kontrol metotlarini barindiran sinif
7:     /// </summary>
8:     public static class TarihIslemleri
9:     {
10:         /// <summary>
11:         /// Filtre ekranlarında başlangıç ve bitis tarihi üzerinden sorgulama
yapildiginda
12:         /// Eger ayni gün için sorgulama yapiliyorsa
13:         /// Bitis tarihini o günün 23.59.59.999'una ayarlamak için
14:         /// Hazirlanan metottur
15:         /// </summary>
16:         /// <param name="dateTime">Tarih Bilgisi</param>
17:         /// <param name="hours">Saat Bilgisi</param>
18:         /// <param name="minutes">Dakika Bilgisi</param>
19:         /// <param name="seconds">Saniye Bilgisi</param>
20:         /// <param name="milliseconds">Mili Saniye Bilgisi</param>
21:         /// <returns>Yeni Tarih Bilgisi</returns>
22:         public static DateTime ZamaniDegistir(this DateTime dateTime, int hours, int
minutes, int seconds, int milliseconds)
23:         {
24:             return new DateTime(
25:                 dateTime.Year,
26:                 dateTime.Month,
27:                 dateTime.Day,
28:                 hours,
29:                 minutes,
30:                 seconds,
31:                 milliseconds,
32:                 dateTime.Kind);
33:         }
34:
35:         /// <summary>
36:         /// Yas hesaplamak için kullanılan metottur
37:         /// </summary>
38:         /// <param name="dogumTarihi">Dogum Tarihi Bilgisi</param>
39:         /// <returns>Yas Bilgisi</returns>
40:         public static int YasHesapla(DateTime dogumTarihi)
41:         {
42:             var now = DateTime.Now;
43:             var years = new DateTime(DateTime.Now.Subtract(dogumTarihi).Ticks).Year -
1;
44:             var pastYearDate = dogumTarihi.AddYears(years);
45:             var months = 0;
46:             for (var i = 1; i <= 12; i++)
47:             {
48:                 if (pastYearDate.AddMonths(i) == now)
49:                 {
50:                     months = i;
51:                     break;
52:                 }
53:
54:                 if (pastYearDate.AddMonths(i) < now) continue;
55:                 months = i - 1;
56:                 break;

```

```

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


```

```

123:         result += " Gün";
124:         return result;
125:     }
126:
127:     /// <summary>
128:     /// Kidem Tazminatı Hesaplaması Yapılırken ayda kaç gün olursa olsun tamamında
çalışmıssa
129:     /// 30 gün üzerinden hesaplanması gerektiği için
130:     /// İsten ayrılış ve iş başlangıç tarihleri arasında kaç gün geçtiğini
belirtmek için
131:     /// Hesaplama yapılmasını sağlayan metottur.
132:     /// </summary>
133:     /// <param name="baslangicTarihi">Başlangıç Tarihi Bilgisi</param>
134:     /// <param name="bitisTarihi">Bitiş Tarihi Bilgisi</param>
135:     /// <returns>Gün Sayısı Bilgisi</returns>
136:     public static int KidemTazminatıTarihAralığıHesapla(DateTime? baslangicTarihi,
DateTime? bitisTarihi)
137:     {
138:         var artacakMi = true;
139:         var result = 0;
140:         var baslangicMi = false;
141:         var bitisMi = false;
142:         var aylıkGünSayısı = 0;
143:         for (var date = (DateTime)baslangicTarihi; date <= (DateTime)bitisTarihi;
date = date.AddDays(1))
144:         {
145:             if (!artacakMi)
146:                 result++;
147:             else
148:                 artacakMi = false;
149:
150:             if (date.Day == 1)
151:                 baslangicMi = true;
152:             else if (date.Day == DateTime.DaysInMonth(date.Year, date.Month))
153:                 bitisMi = true;
154:
155:             if (baslangicMi)
156:                 aylıkGünSayısı++;
157:             if (baslangicMi && bitisMi && aylıkGünSayısı != 30)
158:             {
159:                 result = result - aylıkGünSayısı + 30;
160:                 baslangicMi = false;
161:                 bitisMi = false;
162:                 aylıkGünSayısı = 0;
163:             }
164:             else if (bitisMi)
165:             {
166:                 baslangicMi = false;
167:                 bitisMi = false;
168:                 aylıkGünSayısı = 0;
169:             }
170:         }
171:         if (baslangicTarihi?.Day > bitisTarihi?.Day)
172:             result--;
173:         return result;
174:     }
175:
176:
177:     /// <summary>
178:     /// Girilen iki tarih arasında kaç ay olduğunu bulan metottur
179:     /// </summary>
180:     /// <param name="lValue">İlk tarih bilgisi</param>
181:     /// <param name="rValue">Son tarih bilgisi</param>
182:     /// <returns>Ay Farkı Bilgisi</returns>
183:     public static int AyFarki(DateTime lValue, DateTime rValue) =>
Math.Abs(lValue.Month - rValue.Month + 12 * (lValue.Year - rValue.Year));
184: }
185: }

```

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 63
CSharpHelperMethods.FrmAnaSayfa 32
CSharpHelperMethods.FrmAnaSayfa.btnAyFarki 34
CSharpHelperMethods.FrmAnaSayfa.btnEPostaDogrula 34
CSharpHelperMethods.FrmAnaSayfa.btnIbanDogrula 35
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 36
CSharpHelperMethods.FrmAnaSayfa.btnSayisalMi 36
CSharpHelperMethods.FrmAnaSayfa.btnSha256Hash 36
CSharpHelperMethods.FrmAnaSayfa.btnSha512Hash 36
CSharpHelperMethods.FrmAnaSayfa.btnSifreUygunMu 36
CSharpHelperMethods.FrmAnaSayfa.btnTarihAraligi 36
CSharpHelperMethods.FrmAnaSayfa.btnTcKimlikDogrula 37
CSharpHelperMethods.FrmAnaSayfa.btnTelefonNoDuzenle 37
CSharpHelperMethods.FrmAnaSayfa.btnTurkceKarakterDuzelt 37
CSharpHelperMethods.FrmAnaSayfa.btnYasHesapla 37
CSharpHelperMethods.FrmAnaSayfa.btnYasHesaplaMetinsel 37
CSharpHelperMethods.FrmAnaSayfa.cmbIslemTuru 37
CSharpHelperMethods.FrmAnaSayfa.cmbIslemTuru_SelectedIndexChanged 43
CSharpHelperMethods.FrmAnaSayfa.components 37
CSharpHelperMethods.FrmAnaSayfa.Dispose 51
CSharpHelperMethods.FrmAnaSayfa.dtpBaslangicTarihi 38
CSharpHelperMethods.FrmAnaSayfa.dtpBitisTarihi 38
CSharpHelperMethods.FrmAnaSayfa.FrmAnaSayfa 34
CSharpHelperMethods.FrmAnaSayfa.FrmAnaSayfa_Load 51
CSharpHelperMethods.FrmAnaSayfa.InitializeComponent 51
CSharpHelperMethods.FrmAnaSayfa.label1 38
CSharpHelperMethods.FrmAnaSayfa.label10 38
CSharpHelperMethods.FrmAnaSayfa.label11 38
CSharpHelperMethods.FrmAnaSayfa.label12 38
CSharpHelperMethods.FrmAnaSayfa.label13 38
CSharpHelperMethods.FrmAnaSayfa.label14 39
CSharpHelperMethods.FrmAnaSayfa.label15 39
CSharpHelperMethods.FrmAnaSayfa.label16 39
CSharpHelperMethods.FrmAnaSayfa.label17 39
CSharpHelperMethods.FrmAnaSayfa.label2 39
CSharpHelperMethods.FrmAnaSayfa.label3 39
CSharpHelperMethods.FrmAnaSayfa.label4 40
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 41
CSharpHelperMethods.FrmAnaSayfa.lbSayiSonuc 41
CSharpHelperMethods.FrmAnaSayfa.lbSifreSonuc 41
CSharpHelperMethods.FrmAnaSayfa.lbSonuc 41
CSharpHelperMethods.FrmAnaSayfa.pnlKisilislemleri 41
CSharpHelperMethods.FrmAnaSayfa.pnlMetinIslemleri 41
CSharpHelperMethods.FrmAnaSayfa.pnlSayiIslemleri 42
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 61
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 63	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	
CSharpHelperMethods.YardimciSiniflar.Sifrelemelslemleri 22	
CSharpHelperMethods.YardimciSiniflar.Sifrelemelslemleri.Hes aplaMD5 22	
CSharpHelperMethods.YardimciSiniflar.Sifrelemelslemleri.Hes aplaSHA256 23	
CSharpHelperMethods.YardimciSiniflar.Sifrelemelslemleri.Hes aplaSHA512 23	
CSharpHelperMethods.YardimciSiniflar.Sifrelemelslemleri.Ras tgeleSifreUret 24	
CSharpHelperMethods.YardimciSiniflar.Sifrelemelslemleri.Sifr eUygunMu 24	
CSharpHelperMethods.YardimciSiniflar.Siralamalslemleri 25	
CSharpHelperMethods.YardimciSiniflar.Siralamalslemleri.Chu nkType 25	
CSharpHelperMethods.YardimciSiniflar.Siralamalslemleri.Chu nkType enumeration 25	
CSharpHelperMethods.YardimciSiniflar.Siralamalslemleri.Co mpare 26	
CSharpHelperMethods.YardimciSiniflar.Siralamalslemleri.InC hunk 27	
CSharpHelperMethods.YardimciSiniflar.TarihIslemleri 27	
CSharpHelperMethods.YardimciSiniflar.TarihIslemleri.AyFarki 28	
CSharpHelperMethods.YardimciSiniflar.TarihIslemleri.KidemT azminatiTarihAraligiHesapla 28	
	D
	DataTableIslemleri class 5
	about DataTableIslemleri class 5
	BaseType 5
	Nullable.GetUnderlyingType 5
	oType.GetGenericTypeDefinition 6
	ToDataTable 6
	typeof 7
	DataTableIslemleri.cs 64
	Dosyalslemleri class 7
	about Dosyalslemleri class 7
	DosyaKilitliMi 8
	DosyaUzantisiUygunMu 9
	GetirDosyaUzantisiMimeTypeelle 9
	GetirIcerikTipiDosyaUzantisille 10
	GüvenliDizinOlustur 10
	Dosyalslemleri.cs 66
	F
	Files 62
	FrmAnaSayfa class 32
	about FrmAnaSayfa class 32
	btnAyFarki 34
	btnEPostaDogrula 34
	btnIbanDogrula 35
	btnIlkHarfiBuyukYap 35
	btnIlkHarfleriBuyukYap 35
	btnMd5Hash 35

btnMetinSifrele 35
 btnMetneDonustur 35
 btnOnayKoduOlustur 35
 btnRastgeleSifreUret 36
 btnSayisalMi 36
 btnSha256Hash 36
 btnSha512Hash 36
 btnSifreUygunMu 36
 btnTarihAraligi 36
 btnTcKimlikDogrula 37
 btnTelefonNoDuzenle 37
 btnTurkceKarakterDuzelt 37
 btnYasHesapla 37
 btnYasHesaplaMetinsel 37
 cmbIslemTuru 37
 cmbIslemTuru_SelectedIndexChanged 43
 components 37
 Dispose 51
 dtpBaslangicTarihi 38
 dtpBitisTarihi 38
 FrmAnaSayfa 34
 FrmAnaSayfa_Load 51
 InitializeComponent 51
 label1 38
 label10 38
 label11 38
 label12 38
 label13 38
 label14 39
 label15 39
 label16 39
 label17 39
 label2 39
 label3 39
 label4 40
 label5 40
 label6 40
 label7 40
 label8 40
 label9 40
 lbKisiSonuc 40

lbMetinSonuc 41
 lbSayiSonuc 41
 lbSifreSonuc 41
 lbSonuc 41
 pnlKisilslemleri 41
 pnlMetinIslemleri 41
 pnlSayiIslemleri 42
 pnlSifreIslemleri 42
 pnlTarihIslemleri 42
 txtKisiMetni 42
 txtMetin 42
 txtSayi 42
 txtSifrelenecekMetin 42
 FrmAnaSayfa.cs 68
 FrmAnaSayfa.Designer.cs 77

K

Kisilslemleri class 11
 about Kisilslemleri class 11
 DogrulaEPosta 11
 DogrulaIBAN 12
 DogrulaTcKimlikNo 12
 Kisilslemleri.cs 88

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 90

N

Numeric enumeration member 25

P

Program class 61
 about Program class 61
 Main 62
Program.cs 93

R

Resources class 1
 about Resources class 1
 Culture 3
 resourceCulture 2
 resourceMan 3
 ResourceManager 3
 Resources 2
Resources.Designer.cs 93

S

SayIslemleri class 19
 about SayIslemleri class 19
 AltHaneliOnayKoduOlustur 19
 SayiHalineCevir 20
 SayisalMi 20
 SayiyiYaziyaCevir 21
 text.All 22
SayIslemleri.cs 95
Settings class 3
 about Settings class 3
 Default 4
 defaultInstance 4
Settings.Designer.cs 97
SifrelemIslemleri class 22
 about SifrelemIslemleri class 22
 HesaplaMD5 22
 HesaplaSHA256 23

HesaplaSHA512 23
RastgeleSifreUret 24
SifreUygunMu 24
SifrelemIslemleri.cs 98
SiralamIslemleri class 25
 about SiralamIslemleri class 25
 Compare 26
 InChunk 27
SiralamIslemleri.cs 100

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