






## *Hedefler*

Bu üniteyi çalıştıktan sonra;

-  InpuBox Fonksiyonunu öğrenmek,
-  MessageBox deyimini öğrenmek,
-  File Dialog Pencerelelerini öğrenmek,

### *İçindekiler*

- InputBox Fonksiyonu,
- MessageBox Deyimi,
- OpenFileDialog Kontrolü,
- SaveFileDialog Kontrolü,
- FontDialog Kontrolü,
- ColorDialog Kontrolü,
- FolderBrowserDialog Kontrolü.

## Diyalog Kutuları

### InputBox : Bilgi Giriş Fonksiyonu

**INPUTBOX** fonksiyonu, kullanıcının ekrandan bilgi girişini sağlar. Başlığı olan ve pozisyonu belirlenebilen bir pencere açar. Pencerede kullanıcının bilgi girebileceği bir metin kutusu vardır. Bu fonksiyonun genel formu aşağıdaki şekildedir:

**INPUTBOX** (Kullanıcıya verilecek mesaj, Açılacak formun başlığı, Giriş kutusunda başlangıçta bulunması istenen değer, Giriş formunun sol üst köşesinin X,Y koordinatları)



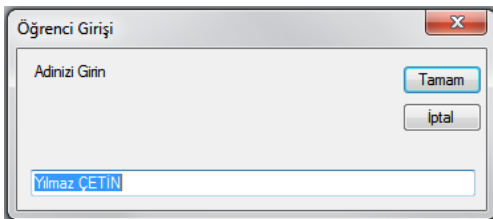
Bu fonksiyonu yazarken bunların hepsinin kullanılması zorunlu değildir.

**Inputbox** içerisinde girilecek olan mesajın uzunluğu en fazla 1024 karakter olabilir.

Örnek : Herhangi bir form üzerinde iken aşağıdaki kod yazılırsa ;

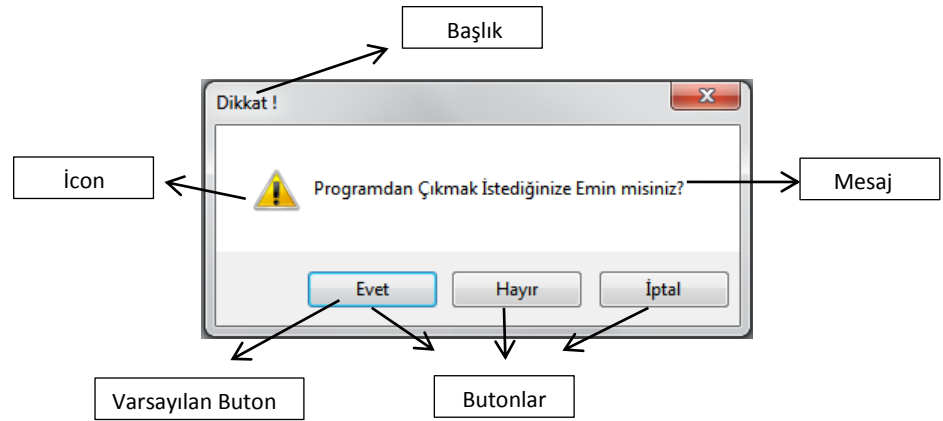
```
ad = InputBox("Adinizi Girin", "Öğrenci Girişi", "Yılmaz ÇETİN", 0, 0)
```

program çalıştırıldığında, form üzerinde aşağıdaki bilgi giriş kutusu çıkacaktır. OK (Tamam) ve Cancel (İptal) butonları komut tarafından otomatik olarak oluşturulmaktadır.



### MessageBox : Mesaj Diyalog Kutusu

**MessageBox** (Mesaj Kutusu) deyimi, program içinde kullanıcıya mesaj vermek amacıyla kullanılmaktadır. Bir mesaj kutusunun bileşenleri şu şekildedir;



**MessageBox** diyalog kutusu program içinde, **Show** metodu ile kullanıcıya gösterilir. Bu diyalog kutusunun birkaç kullanım şekli vardır:

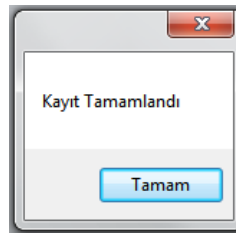
1. En basit şekli: Kullanıcıya mesaj vermek. Bu durumda **MessageBox** deyimini şu şekilde yazılır ;

```
MessageBox.Show(Kullanıcıya verilecek mesaj)
```

**Örnek :** Herhangi bir form üzerinde iken aşağıdaki kod yazılırsa;

```
MessageBox.Show("Kayıt Tamamlandı")
```

program çalıştırıldığında, form üzerinde aşağıdaki mesaj kutusu çıkacaktır. OK (Tamam) butonu komut tarafından otomatik olarak oluşturulmaktadır.



2. En genel kullanımı:

```
MessageBox.Show(Kullanıcıya verilecek mesaj, Form Başlığı, Butonlar, İkon, Varsayılan botun)
```

**Butonlar :** Bu kısımda **MessageBoxButtons** sabitleri yazılabilir. Bunlar

```
MessageBoxButtons.değer
```

şeklinde yazılırlar.

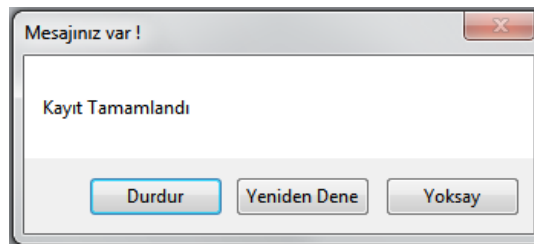
Sabit Değerler	Diyalog Kutusundaki Görünüm
OK	
OKCancel	 
AbortRetryIgnore	  
YesNoCancel	  
YesNo	 
RetryCancel	 

Windows' un Türkçe veya İngilizce olmasına göre butonlar Türkçe veya İngilizce olur. Örneğin Türkçe Windows işletim sisteminde, **OK** yerine **Tamam**, **Cancel** yerine **İptal** yazılı butonlar gözükecektir.

**Örnek :** Herhangi bir form üzerinde iken aşağıdaki kod yazılırsa;

```
MessageBox.Show("Kayıt Tamamlandı", "Mesajınız var !", MessageBoxButtons.AbortRetryIgnore)
```









program çalıştırıldığında, form üzerinde aşağıdaki mesaj kutusu çıkacaktır.



**İkon:** Bu kısımda ilgili kod yazılabileceği gibi, **MessageBoxIcon** sabitleri de yazılabilir. Bunlar;

**MessageBoxIcon.değer**

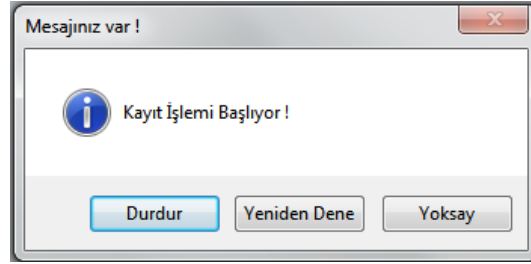
şeklinde yazılabilir.

Sabit Değerler	Diyalog Kutusundaki Görünüm
Asterisk	
Error	
Exclamation	
Hand	
Information	
None	
Question	
Stop	
Warning	

**Örnek :** Herhangi bir form üzerinde iken aşağıdaki kod yazılırsa;

```
MessageBox.Show("Kayıt İşlemi Başlıyor !", "Mesajınız var !",
MessageBoxButtons.AbortRetryIgnore, MessageBoxIcon.Information)
```

program çalıştırıldığında, form üzerinde aşağıdaki mesaj kutusu çıkacaktır.



Not : `MessageBox.Show` yerine `Msgbox` da kullanılabilir. Örneğin yukarıda örnekteki deyim yerine aşağıdaki deyim de yazılabilir.

```
MsgBox("Kayıt İşlemi Başlıyor !", MsgBoxStyle.AbortRetryIgnore +
MsgBoxStyle.Information, "Mesajınız var !")
```

**Varsayılan Buton :** Bu kısımda ilgili kod yazılabileceği gibi, `MessageBoxDefaultButton` sabitleri de yazılabilir. Bunlar ;

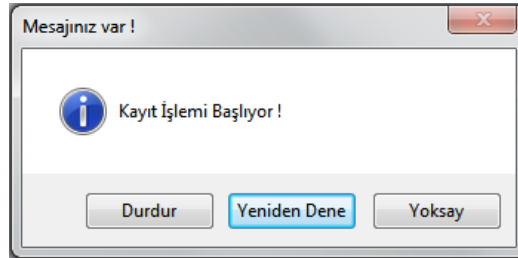
`MessageBoxDefaultButton.değer`

şeklinde yazılabilir.

Sabit Değerler	Varsayılan Buton
Button1	Birinci Buton
Button2	İkinci Buton
Button3	Üçüncü Buton

**Örnek :** Herhangi bir form üzerinde iken aşağıdaki kod yazılırsa;

```
MessageBox.Show("Kayıt İşlemi Başlıyor !", "Mesajınız var !",
MessageBoxButtons.AbortRetryIgnore, MessageBoxIcon.Information,
MessageBoxDefaultButton.Button2)
```





**MessageBox** fonksiyonundaki butonlardan biri tıklandığında, bu fonksiyon bir değer döndürecek. Bu değer bir kod ile belirleneceği gibi, **DialogResult.değer** özelliği ile de sabit bir değer döndürülebilir.

## FILE DIALOG PENCERELERİ

Diyalog pencereleri **Toolbox** içerisinde **Dialogs** başlığı altında bulunur. OpenFileDialog, SaveFileDialog, FontDialog, ColorDialog ve FolderBrowserDialog olmak üzere beş adet dialog kutusu vardır.

### OpenFileDialog : Dosya Açma Diyalog Kutusu

**OpenFileDialog**, herhangi bir Windows uygulamasında File → Open veya Dosya → Aç komutlarında ekrana gelen Dosya Açma arayüzünü uygulamalarda kullanmamızı sağlayan bir komuttur. Toolbox üzerinden seçtiğimiz **OpenFileDialog** kontrolü, form üstüne bırakılsa bile, formun altında görüntülenecektir.

Toolbox	Form Altındaki Görünümü
 OpenFileDialog1	 OpenFileDialog1

Properties (Özellikler)	
<b>Name</b>	<b>OpenFileDialog</b> un ismi.
<b>AddExtension</b>	Kullanıcı ihmal etse bile, dosya adına bir uzantının otomatik olarak eklenip eklenmeyeceğini belirler.
<b>CheckFileExists</b>	Kullanıcının belirlediği dosyanın bulunmaması durumunda, bir uyarı mesajının gösterilip gösterilmeyeceğini belirler.
<b>CheckPathExists</b>	Kullanıcının belirlediği yolun bulunmaması durumunda, bir uyarı mesajının gösterilip gösterilmeyeceğini belirler.
<b>DefaultExt</b>	Varsayılan dosya uzantısını belirler.
<b>FileName</b>	Diyalog kutusunda seçilen dosya ismini bir string olarak alır (saklar).
<b>Filter</b>	"Files of Type" "Dosya Tipi" için seçenek belirler. Buraya yazacağımız seçeneklerdeki dosya türleri listelenir. Örneğin "doc" yazılırsa, yalnızca Word dosyaları listelenir.
<b>InitialDirectory</b>	Diyalog kutusu tarafından başlangıçta görüntülenecek klasörü belirler.
<b>Title</b>	Diyalog Kutusu başlığı

Events (Olaylar)	
<b>ShowDialog</b>	Diyalog Kutusunu gösterir.

**ShowDialog** metodu **OpenFileDialog** kontrolünü göstermek için kullanılır.

**Örnek :**

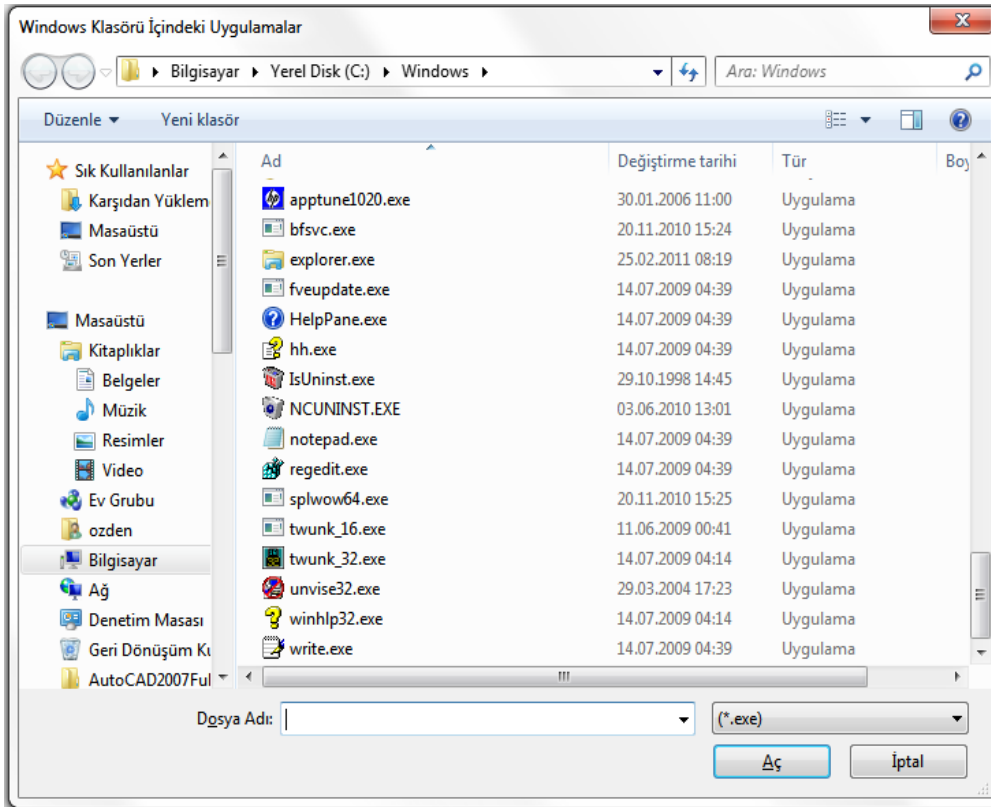
<b>Name</b>	: dlgDosyaAc
<b>Filter</b>	: Uygulama Dosyaları (*.exe) *.exe
<b>InitialDirectory</b>	: C:\Windows
<b>Title</b>	: Windows Klasörü İçindeki Uygulamalar

olarak belirlemiş olalım. Herhangi bir butonun tıklanması (yada başka bir olay olabilir) durumunda, kod sayfasında Dosya Açma Diyalog Kutusunu ekrana getirmek için ;

```
dlgDosyaAc.ShowDialog()
```

yazıyoruz. Bu satırın çalışması ile ekranda aşağıdaki diyalog kutusu görülecektir.





Bu diyalog kutusunda işlemiz bitince yapabileceğimiz iki işlem vardır. Ya işlemi iptal etmek için **Cancel** (İptal) butonuna basacağız, bu durumda programa **DialogResult.Cancel** değeri dönecektir. Yada bir dosya seçip, **Open** (Aç) butonuna basacağız, o zaman programa **DialogResult.Ok** değeri dönecektir ve seçilen dosya tüm klasör ismi ile beraber **FileName** özelliğine aktarılacaktır.

### SaveFileDialog : Kaydetme Diyalog Kutusu

**SaveFileDialog**, herhangi bir Windows uygulamasında File → Save veya Dosya → Kaydet komutlarında ekrana gelen Dosya Kaydetme arayüzünü uygulamalarda kullanmamızı sağlayan bir kontroldür. **Toolbox** üzerinden seçtiğimiz **SaveFileDialog** kontrolü, form üstüne bırakılsa bile, **OpenFileDialog** gibi formun altında görüntülenecektir.



Properties (Özellikler)	
<b>Name</b>	<i>SaveFileDialog</i> ' un ismi.
<b>AddExtension</b>	Kullanıcı ihmal etse bile, dosya adına bir uzantının otomatik olarak eklenip eklenmeyeceğini belirler.
<b>CheckFileExists</b>	Kullanıcının belirlediği dosyanın bulunmaması durumunda, bir uyarı mesajının gösterilip gösterilmeyeceğini belirler. Özellikle dosyanın aynı isimli bir başka dosya üzerine kaydedilmesini engelleme açısından faydalıdır.
<b>CheckPathExists</b>	Kullanıcının belirlediği yolun bulunmaması durumunda, bir uyarı mesajının gösterilip gösterilmeyeceğini belirler.
<b>CheckPrompt</b>	Kullanıcının belirlediği dosyanın oluşturulması izni için diyalog kutusunun gösterilip gösterilmeyeceğini belirler.
<b>DefaultExt</b>	Varsayılan dosya uzantısını belirler.
<b>FileName</b>	Diyalog kutusunda seçilen dosya ismini bir string olarak alır (saklar).
<b>Filter</b>	"Files of Type" "Dosya Tipi" için seçenek belirler.
<b>InitialDirectory</b>	Diyalog kutusu tarafından başlangıçta görüntülenecek klasörü belirler.
<b>OverwritePrompt</b>	Mevcut bir dosya üzerine yazma durumunda uyarı diyalog kutusunun çıkıp çıkmayacağını belirler.
<b>Title</b>	Diyalog Kutusu başlığı

Events (Olaylar)	
<b>ShowDialog</b>	Diyalog Kutusunu gösterir.

*ShowDialog* metodu *SaveFileDialog* kontrolünü göstermek için kullanılır.

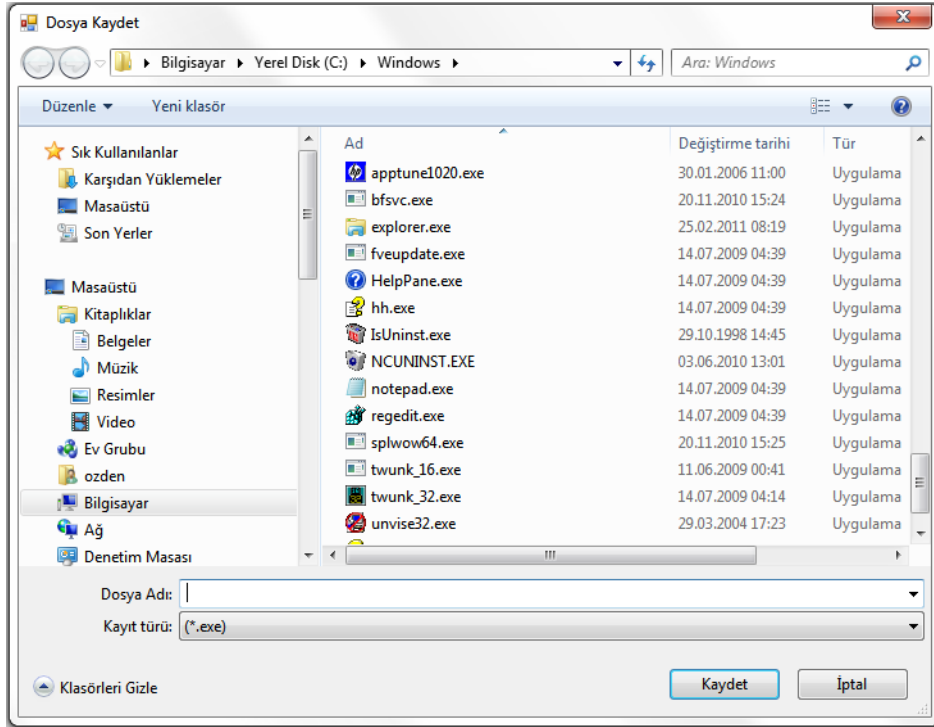
**Örnek :**

<b>Name</b>	: dlgDosyaKaydet
<b>Filter</b>	: Uygulama Dosyaları (*.exe) *.exe
<b>InitialDirectory</b>	: C:\Windows
<b>OverwritePrompt</b>	: True
<b>Title</b>	: Dosya Kaydet

olarak belirlemiş olalım. Herhangi bir butonun tıklanması (yada başka bir olay olabilir) durumunda, kod sayfasında Dosya Kaydet Diyalog Kutusunu ekrana getirmek için ;

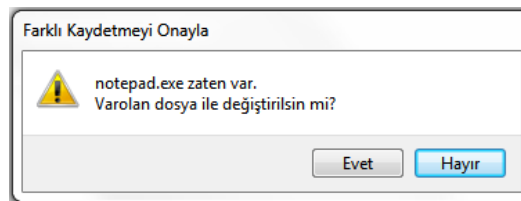
```
dlgDosyaKaydet.ShowDialog()
```

yazıyoruz. Bu satırın çalışması ile ekranda aşağıdaki diyalog kutusu gözükcektir.



Bu diyalog kutusunda işlem bitince yapabileceğimiz iki işlem vardır. Ya işlemi iptal etmek için **Cancel** (İptal) butonuna basacağız, bu durumda programa **DialogResult.Cancel** değeri dönecektir. Yada bir dosya seçip, **Save** (Kaydet) butonuna basacağız, o zaman programa **DialogResult.Ok** değeri dönecektir ve seçilen dosya tüm klasör ismi ile beraber **FileName** özelliğine aktarılacaktır.

Eğer yazdığımız dosya ismi mevcut dosya isimlerinden birisi ise;



Şeklinde uyarı diyalog kutusu çıkacaktır.

## FontDialog : Yazı Tipi Diyalog Kutusu

**FontDialog**, geliştirdiğimiz uygulamalarda yazı tipini belirlememizi sağlayan kontroldür. Bu diyalog kontrolü de diğer kontroller gibi formun altında görüntülenir.



Properties (Özellikler)	
<b>Name</b>	<b>FontDialog</b> ' un ismi.
<b>Color</b>	Seçilen yazı rengi
<b>Font</b>	Seçilen font.
<b>MaxSize</b>	Kullanıcının seçebileceği maksimum font büyüklüğü.
<b>MinSize</b>	Kullanıcının seçebileceği minimum font büyüklüğü.
<b>ShowColor</b>	Diyalog kutusunda yazı rengi seçiminin gösterilip gösterilmemesini belirler.
<b>ShowEffects</b>	Diyalog kutusunda fonta ilgili çeşitli özelliklerin (üstü çizili, altı çizili gibi) gösterilip, gösterilmemesini belirler.

Events (Olaylar)	
<b>ShowDialog</b>	Diyalog Kutusunu gösterir.

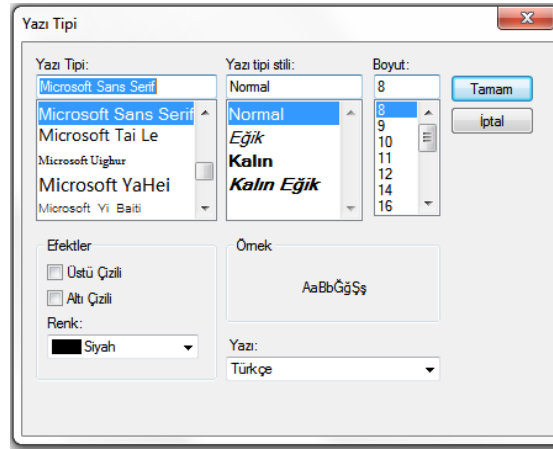
**ShowDialog** metodu **FontDialog** kontrolünü göstermek için kullanılır.

**Örnek :**    **Name**                : dlgFont1  
                   **ShowColor**       : True  
                   **ShowEffects**    : True

olarak belirlemiş olalım. Herhangi bir butonun tıklanması (yada başka bir olay olabilir) durumunda, kod sayfasında Font Diyalog Kutusunu ekrana getirmek için ;

```
dlgFont1.ShowDialog()
```

yazıyoruz. Bu satırın çalışması ile ekranda aşağıdaki diyalog kutusu gözükcektir.



Bu diyalog kutusunda işlemiz bitince yapabileceğimiz iki işlem vardır. Ya işlemi iptal etmek için **Cancel** (İptal) butonuna basacağız, bu durumda programa **DialogResult.Cancel** değeri dönecektir. Yada font, color vb. seçip, **OK** (Tamam) butonuna basacağız, o zaman programa **DialogResult.Ok** değeri dönecektir ve seçilen dosya tüm font, color bilgileri **Font** ve **Color** özelliğine aktarılacaktır.

## ColorDialog : Renk Diyalog Kutusu

**ColorDialog**, geliştirdiğimiz uygulamalarda renk belirlememizi sağlayan kontroldür. Belirlediğimiz rengi kontrol özelliklerinde (buyon rengi, formun zemin rengi vb.) kullanacağımız gibi, grafik uygulamalarında da kullanabiliriz.

Toolbox	Form Altındaki Görünümü

Properties (Özellikler)	
<b>Name</b>	<b>ColorDialog'</b> un ismi.
<b>AllowFullOpen</b>	Kullanıcının özel renklerini tanımlamasına ( <b>Custom Colors</b> ) izin verilip verilmemesini belirler.
<b>AnyColor</b>	<b>Basic Colors</b> (Temel Renkler) setinde tüm renklerin gösterilip gösterilmeyeceğini belirler.
<b>Color</b>	Seçilen renk.
<b>FullOpen</b>	<b>Color</b> diyalog kutusu açıldığında, <b>Custom Colors</b> kontrolünün olup olmayacağını belirler. <b>AllowFullOpen</b> değeri <b>False</b> ise bu özelliğin bir önemi yoktur.
<b>SolidColorOnly</b>	<b>True</b> olması durumunda kullanıcı yalnızca <b>Solid</b> renkleri seçebilmekle kısıtlanacaktır.

Events (Olaylar)	
<b>ShowDialog</b>	Diyalog Kutusunu gösterir.

**ShowDialog** metodu **ColorDialog** kontrolünü göstermek için kullanılır.

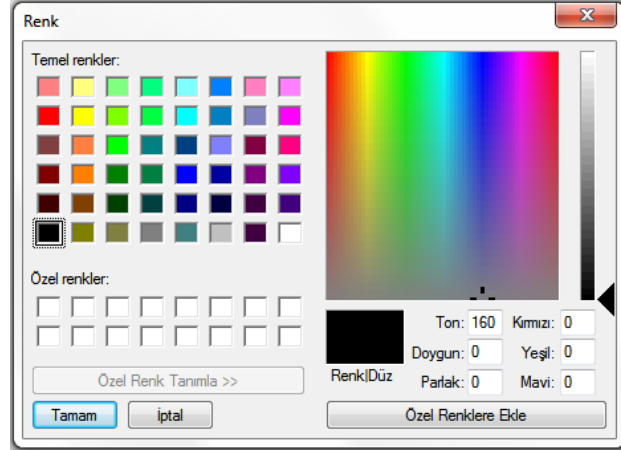
**Örnek :**

<b>Name</b>	: dlgColor1
<b>AllowFullOpen</b>	: True
<b>AnyColors</b>	: True
<b>FullOpen</b>	: True

olarak belirlemiş olalım. Herhangi bir butonun tıklanması (yada başka bir olay olabilir) durumunda, kod sayfasında Font Diyalog Kutusunu ekrana getirmek için ;

```
dlgColor1.ShowDialog()
```

yazıyoruz. Bu satırın çalışması ile ekranda aşağıdaki diyalog kutusu gözükecektir.



Bu diyalog kutusunda işimiz bitince yapabileceğimiz iki işlem vardır. Ya işlemi iptal etmek için **Cancel** (İptal) butonuna basacağız, bu durumda programa **DialogResult.Cancel** değeri dönecektir. Yada renk seçip, **OK** (Tamam) butonuna basacağız, o zaman programa **DialogResult.Ok** değeri dönecektir ve seçilen dosya tüm Color (renk) bilgileri **Color** özelliğine aktarılacaktır.

## FolderBrowserDialog : Klasöre Gözet Diyalog Kutusu

**FolderBrowserDialog**, geliştirdiğimiz uygulamalarda yedekleme işlemi, geçici işlemler için bir klasör yada işleme tabi tutmak için dosyaları tarayacağı klasör bilgisi belirlemede kullanılır. Kullanıcılar **FolderBrowserDialog** diyalog kutusu ile mevcut diskleri ve disk içerisindeki klasör yapısını görebilecekleri gibi, ayrıca istedikleri bir konumda klasör de oluşturabilirler.

Toolbox	Form Altındaki Görünümü
 FolderBrowserDialog	 FolderBrowserDialog1

Properties (Özellikler)	
<b>Name</b>	<b>FolderBrowserDialog'</b> un ismi.
<b>Description</b>	Diyalog kutusunda açıklayıcı bir mesaj göstermek için kullanılır
<b>RootFolder</b>	Diyalog kutusunun başlangıç konumunu belirler.
<b>SelectedPath</b>	Diyalog kutusunda seçilen klasörün tam yolunu verir.
<b>ShowNewFolderButton</b>	Yeni klasör açma işlemini gerçekleştirecek butonun gösterilip gösterilmemesini belirler.

Events (Olaylar)	
<b>ShowDialog</b>	Diyalog Kutusunu gösterir.

**ShowDialog** metodu **FolderBrowserDialog** kontrolünü göstermek için kullanılır.

**Örnek :** **Name** : dlgGozat  
**Description** : Burada Windows Klasörü Görüntüleniyor  
**SelectedPath** : C:\Windows

olarak belirlemiş olalım. Herhangi bir butonun tıklanması (yada başka bir olay olabilir) durumunda, kod sayfasında Font Diyalog Kutusunu ekrana getirmek için ;

```
dlgGozat.ShowDialog()
```

yazıyoruz. Bu satırın çalışması ile ekranda aşağıdaki diyalog kutusu görülecektir.

