

Nesneye Dayalı Programlama

Sakarya Üniversitesi
Bilgisayar ve Bilişim Bilimleri Fakültesi
Bilgisayar Mühendisliği

Prof. Dr. Ümit KOCABIÇAK
Prof. Dr. Cemil Öz
Öğr. Gör. Nevzat TAŞBAŞI
Öğr. Gör. Sinan İLYAS

2018
3. HAFTA

string ve char[]

string ile char [] birbirinden farklıdır

```
string str = "merhaba"; // DERLENİR
char[] chrArr1 = "1234"; // DERLENMEZ
char[] chrArr2 = {'1','2','3','4'}; // DERLENİR
char[] chrArr3 = str.ToCharArray(); // DERLENİR
char c1 = str[1] // DERLENİR (c1=e)
str[1] = 'a'; // DERLENMEZ, SALT OKUNUR
char c2 = str[0]; // DERLENİR (c2=m)
```

YAYGIN string METOTLARI

- `string.Compare()` – iki string'i karşılaştırır
- `string.Format()` – string'i biçimlendirir
- `string.ToCharArray()` – char dizisine çevirir
- `string.IndexOf()` – string içinde arama yapar
- `string.ToUpper()` – büyük harfe çevirir
- `string.ToLower()` – küçük harfe çevirir
- `string.Concat()` – iki string'i birleştirir
- `string.Join()` – verilen string dizisinin elemanlarını birleştirir
- `string.Remove()` – string'in bir kısmını çıkarır
- `string.LastIndexOf()` – string içinde arama yapar
- `string.Replace()` – bul-değiştir işlemi yapar
- `string.Trim()` – baştaki ve sonraki beyaz boşlukları siler
- `string.Equals()` – iki string aynı mı diye karşılaştırır
- `string.Substring()` – string'in içinden bir kısmını alır
- `string.split()` – string'i string dizisine böler
- `string.StartsWith()` – başta belirli bir string var mı kontrol eder

NUMARALANDIRMA (enum)

```
enum Renk { Mavi, Kirmizi, Yesil };
```

```
•
```

```
•
```

```
•
```

```
Renk r1 = Renk.Mavi;
```

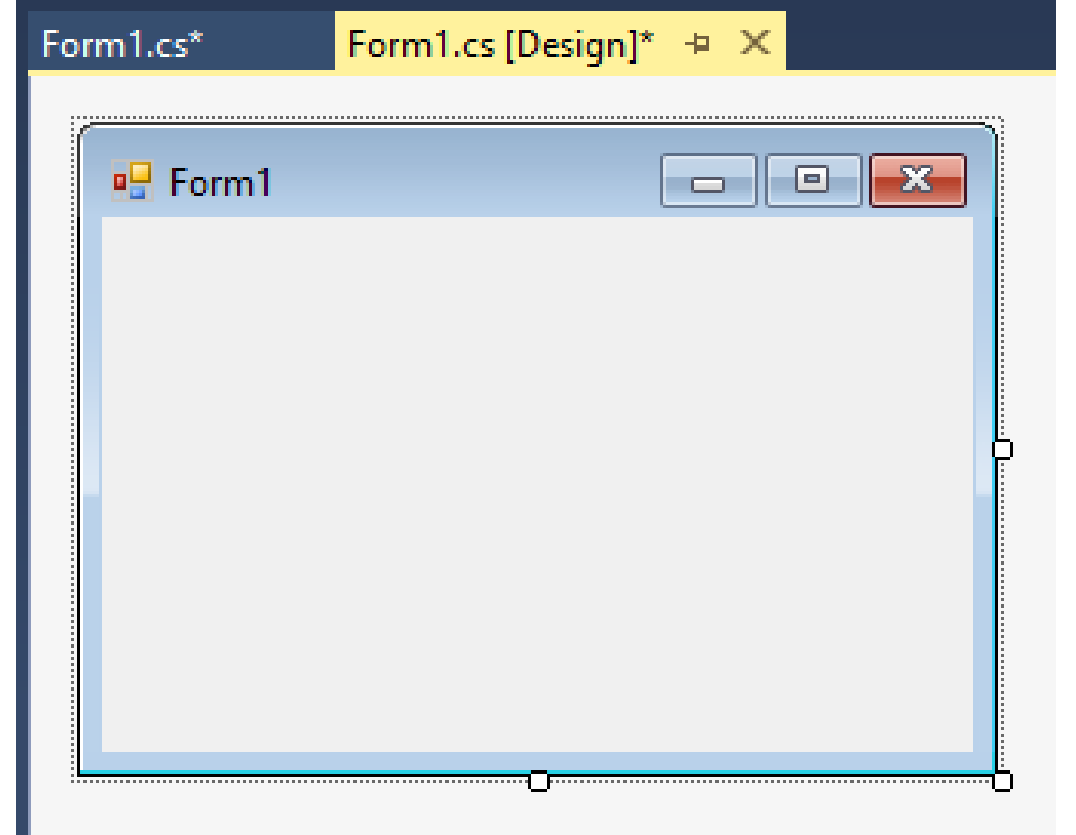
```
if (r1 == Renk.Mavi) Console.WriteLine("Mavi renkli.");
```

WINDOWS FORMS UYGULAMALARI

Bir Windows Forms uygulaması geliřtirmek için Visual Studio'da New Project penceresinde Windows Forms Application seçilir.

Windows pencerelerine programlamada Form adı verilir.

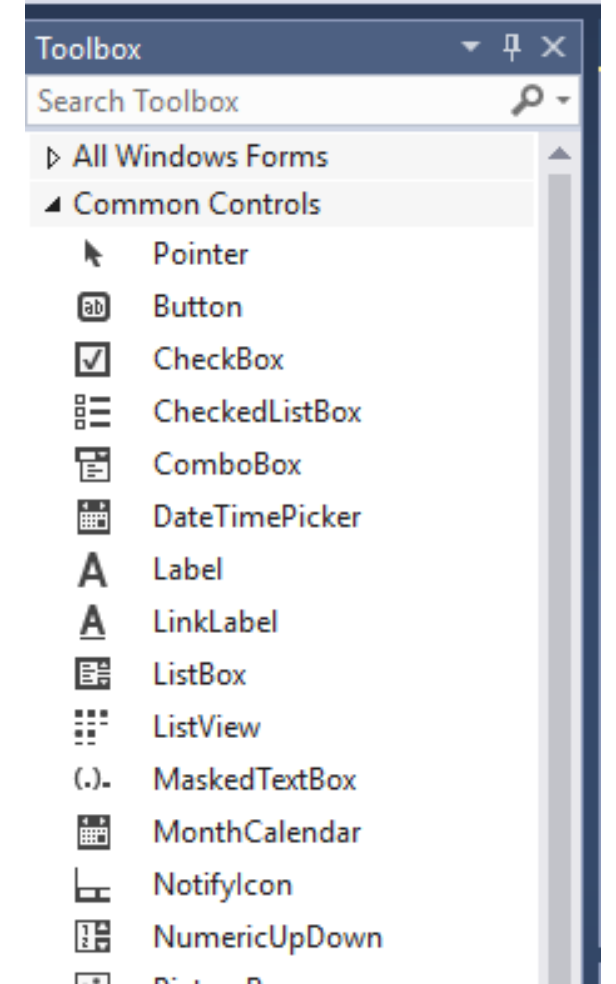
Yeni proje oluşturulduğunda, boş bir form kendiliğinden oluşturulur. İhtiyaç halinde projeye başka formlar da eklenebilir.



KONTROL

Formların üzerine yerleştirilebilen düğme, metin kutusu, vs. gibi her türlü nesneye kontrol adı verilir.

Kontroller Toolbox penceresinde yer alır. Bu pencere görünür değilse View menüsünden Toolbox tıklanarak açılabilir.

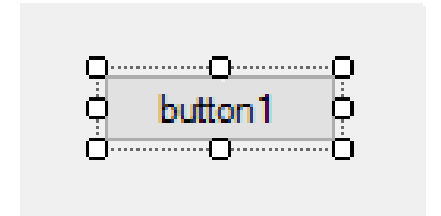


FORM TASARIMI

Formun üzerine bir kontrol yerleřtirmek için üzerine çift tıklanabilir veya sürükleyip formun üzerine bırakılabilir.

Bir kontrolün form üzerindeki yerini deęiřtirmek için fare ile sürükleyerek taşınabilir.

Bir kontrolü yeniden boyutlandırmak için fare ile tek tıklayıp seçtikten sonra kenar ve köşelerinde beliren yeniden boyutlandırma noktalarından tutup çekilerek boyutu deęiřtirilebilir.



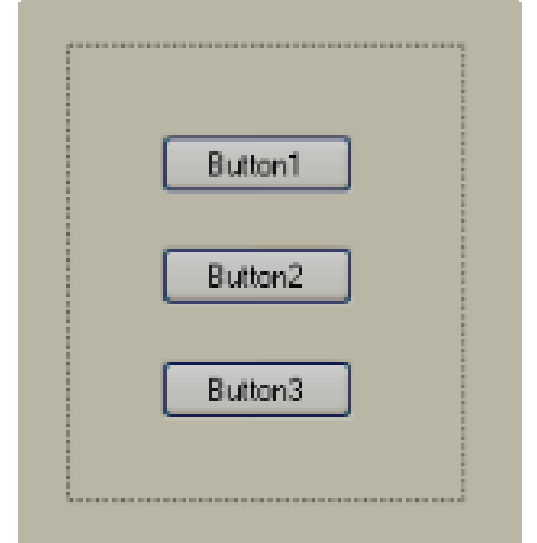
FORM TASARIMI

Birden fazla kontrolü seçmek için form üzerinde fare ile seçilmek istenen kontrolleri içine alan bir dikdörtgen çizilir.

İkinci bir yöntem ise Ctrl veya Shift tuşu ile birlikte seçilmek istenen kontrollere tek tek tıklamaktır.

Formun boş bir yerine tek tıklayarak seçim iptal edilebilir.

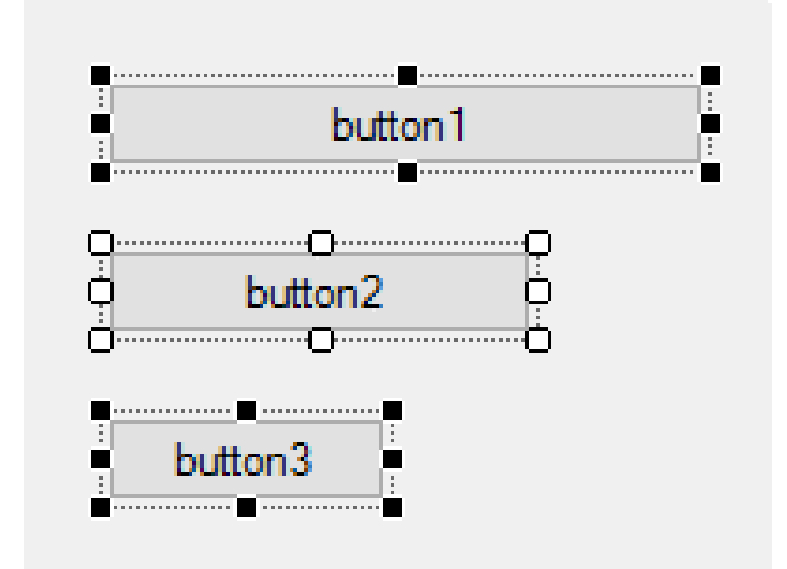
Birden fazla kontrol seçildiğinde, properties penceresinde bu kontrollerin ortak özellikleri görüntülenir. Bu özelliklerden birinde güncelleme yapıldığında, her iki kontrolün özelliği birden güncellenmiş olur.



FORM TASARIMI

Birden fazla kontrol seçildiğinde, bunlardan bir tanesinin etrafındaki noktalar beyaz, diğerlerinin siyah olur. Beyaz olan kontrol, seçili olanların içinde öncelikli olandır. Bütün işlemler bu kontrol baz alınarak yapılır. Örneğin, Layout araç çubuğundan «Make Same Width» (genişliklerini eşitle) komutu verilirse, diğer bütün kontrollerin genişliği beyaz olanın genişliği ile aynı yapılır.

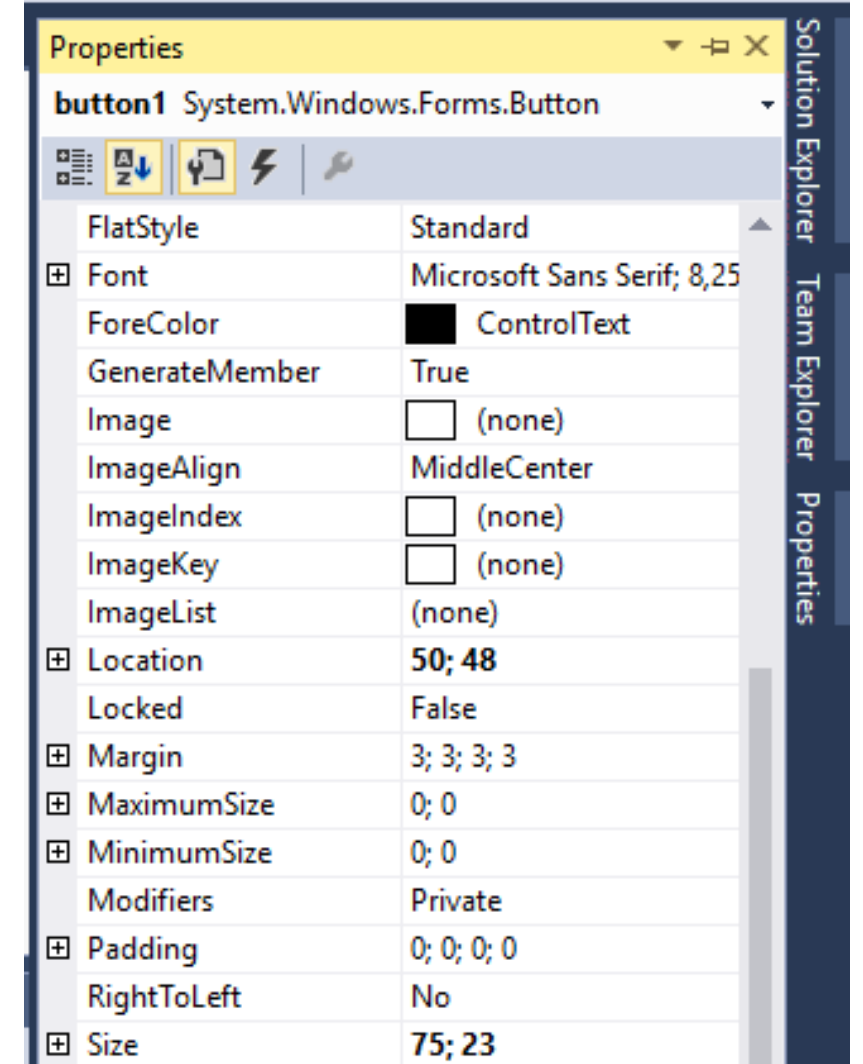
Bir seçimde başka bir kontrolü öncelikli hale getirmek için o kontrolün üzerine tek tıklamak yeterlidir.



ÖZELLİKLER

Kontrollerin belirli özellikleri vardır. Bu özellikler Properties penceresinden değiştirilebilir.

Eğer Properties penceresi açık değilse View menüsünden Properties tıklanarak açılabilir. Ya da herhangi bir kontrolün veya formun üzerine sağ tıklayıp Properties komutu verilerek de açılabilir.



ÖZELLİKLER

Kontrollerin özellikleri kod yazarak değiştirilmek istenirse, kontrolün adının ardından nokta koyulduktan sonra özelliğin adı yazılır.

```
cikisButonu.Text = "ÇIKIŞ";
```

```
cikisButonu.BackColor = Color.Blue;
```

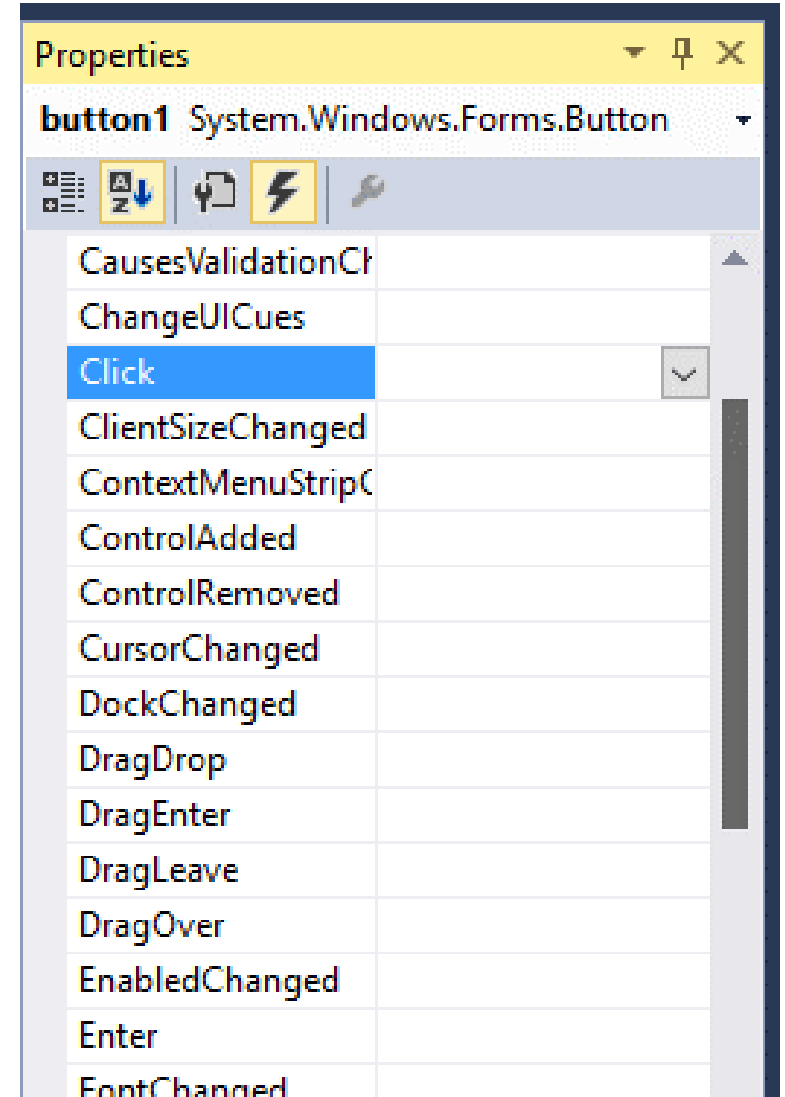
OLAYLAR

Her kontrolün başına gelebilecek belirli olaylar vardır. Kontrole ait olaylar listesine Properties penceresindeki şimşek simgeli butona basarak erişilebilir.

Herhangi bir olay gerçekleştiğinde çalışması istenen kodlar, o olayla ilişkilendirilen metodun içerisine yazılır.

Olay listesinde herhangi bir olayın üzerine çift tıklandığında, otomatik olarak bu işlem için uygun bir metod yazılır ve kontrolün o olayı ile ilişkilendirilir.

Programcı önceden kendi yazdığı bir metodu elle kendisi de bir kontrolün bir olayı ile ilişkilendirebilir. Ancak bir metodun bir olayla ilişkilendirilebilmesi için, o olaya uygun parametrelere sahip olması gerekir.

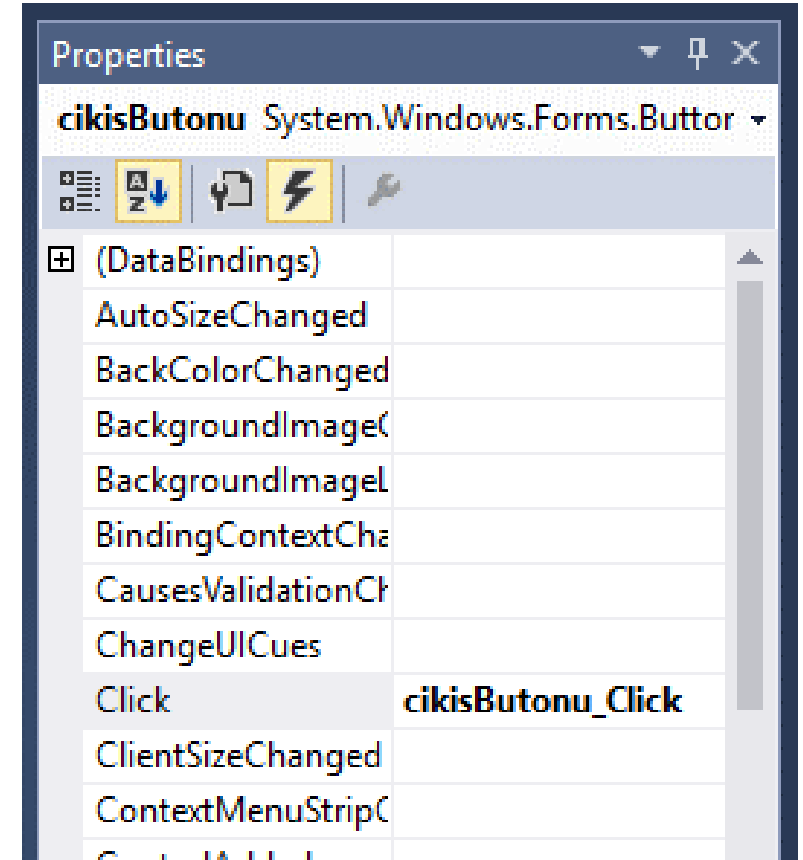


OLAYLAR

cikisButonu isimli butona tıklandığında çalışması istenen kodlar otomatik olarak oluşturulmuş olan aşağıdaki metodun içerisine yazılır.

1 reference

```
private void cikisButonu_Click(object sender, EventArgs e)
{
}
}
```

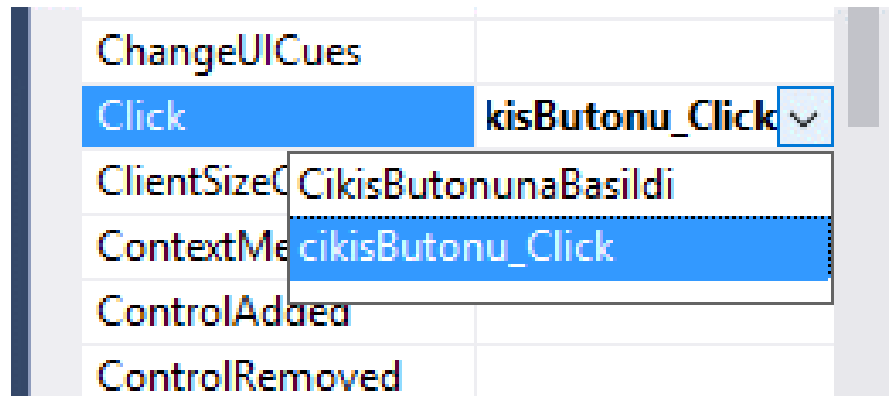


OLAYLAR

Elle yazılmış aşağıdaki metot, istenen olayla elle ilişkilendirilebilir.

1 reference

```
private void CikisButonunaBasildi(object sender, EventArgs e)
{
    Close();
}
```

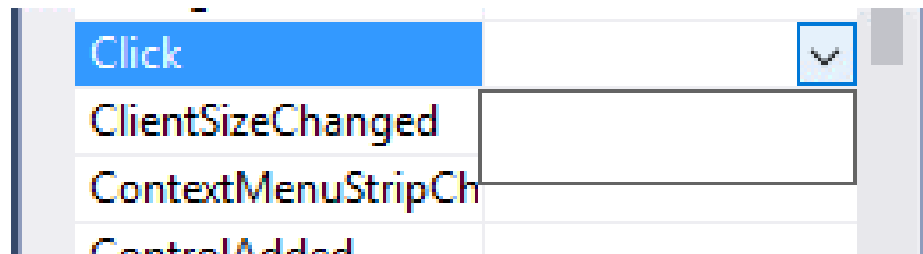


OLAYLAR

Elle yazılmış aşağıdaki metot, uygun parametrelerle yazılmadığı için olaylarla ilişkilendirilemez.

0 references

```
private void CikisButonunaBasildi()  
{  
    Close();  
}
```



METOTLAR

Kontrollerin kendilerine özel hazır metotları vardır. Bu metotlar çağrıldığında kontrol belirtilen işi yapar.

Kontrollere ait metotlar kontrolün adının ardından nokta koyulduktan sonra metodun adı yazılarak çağrılır.

```
cikisButonu.PerformClick();
```

Yukarıdaki metot çağrıldığında, kullanıcı cikisButonu adlı kontrole tıklamış gibi işlem yapılır.

```
cikisButonu.Focus();
```

Yukarıdaki metot çağrıldığında, imleç cikisButonu adlı kontrolün üzerine getirilerek o kontrol aktif kontrol haline getirilir.

ORTAK ÖZELLİKLER

(Name)

Kontrolün değişken adıdır. Kontrole koddan bu ad kullanılarak erişilir.

AutoSize

Eğer bu özellik True olursa, kontrolün üzerine yazılan yazıya göre kontrolün boyutları otomatik olarak ayarlanır.

BackColor

Kontrolün arka plan rengi.

BorderStyle

Kontrolün kenar çizgi stili.

None: Kenar çizgisi yok

Fixed3D: 3 boyutlu kenar çizgisi

FixedSingle: Düz çizgiden oluşan kenar çizgisi

ORTAK ÖZELLİKLER

Enabled

False olursa kontrol kullanılamaz (fare ile tıklanamaz ve değiştirilemez) hale gelir.

Font

Yazı tipi

ForeColor

Kontrolün üzerindeki yazının rengi

Location

Kontrolün formun üzerindeki konumu

☐	Location	48; 39
	X	48
	Y	39

Locked

True olursa tasarım sırasında kontrolün konum ve boyutunun değiştirilmesini engeller.

ORTAK ÖZELLİKLER

MaximumSize

Formun veya kontrolün yeniden boyutlandırılırken alabileceği en büyük genişlik ve yükseklik değerleri.

MinimumSize

Formun veya kontrolün yeniden boyutlandırılırken alabileceği en küçük genişlik ve yükseklik değerleri.

Size

Kontrolün boyutu

Size	75; 23
Width	75
Height	23

TextAlign

Yazının kontrol üzerindeki konumu

TextAlign	MiddleCenter	▼
TextImage	<input type="checkbox"/>	<input type="checkbox"/>
UseCompound	<input type="checkbox"/>	<input type="checkbox"/>
UseMnemonic	<input type="checkbox"/>	<input type="checkbox"/>
UseVisualStyle	<input type="checkbox"/>	<input type="checkbox"/>
UseWaitCursor	<input type="checkbox"/>	<input type="checkbox"/>
Visible	<input type="checkbox"/>	<input type="checkbox"/>

ORTAK ÖZELLİKLER

Visible

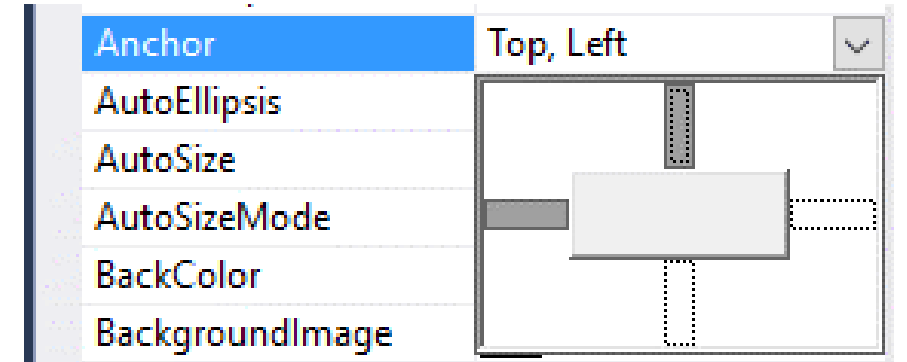
False olarak ayarlanırsa kontrol görünmez olur.

Anchor

Form yeniden boyutlandırıldığında kontrolün formun hangi kenarlarına olan mesafesini koruyacağı bu özellikten ayarlanır.

Eğer «Bottom; Right» şeklinde ayarlanmışsa form büyüdükçe kontrol aşağıda ve sağa doğru kayar. Küçüldükçe yukarı ve sola doğru kayar.

Eğer «Left; Right» şeklinde ayarlanmışsa form genişledikçe kontrolün de genişliği büyür, küçüldükçe kontrolün de genişliği küçülür.



ORTAK OLAYLAR

BackColorChanged

Kontrolün arka plan rengi değiştirildiğinde çalışır.

Click

Kontrolün üzerine fare ile tek tıklandığında veya buton seçiliyken klavyeden ENTER tuşuna basıldığında çalışır.

DragDrop

Sürükle-bırak işlemi tamamlandığında çalışır.

EnabledChanged

Enabled özelliği değiştirildiğinde çalışır.

Enter

Kontrol formdaki aktif kontrol olduğunda çalışır.

ORTAK OLAYLAR

FontChanged

Yazı tipi değiştirildiğinde çalışır.

ForeColorChanged

Metin rengi değiştirildiğinde çalışır.

KeyDown

Kontrolün üzerindeyken klavyeden herhangi bir tuşa basıldığında çalışır.

KeyPress

Kontrolün üzerindeyken klavyeden herhangi bir tuşa basılıp çekildiğinde çalışır.

KeyUp

Kontrolün üzerindeyken klavyeden basılı olan tuş serbest bırakıldığında çalışır.

ORTAK OLAYLAR

Leave

Kontrol formun üzerindeki aktif kontrolken pasif duruma düştüğünde çalışır.

LocationChanged

Kontrolün konumu değiştirildiğinde çalışır.

MouseClicked

Fare ile tıklandığında çalışır.

MouseDown

Kontrolün üzerindeyken farenin bir tuşuna basıldığında çalışır.

MouseEnter

Fare imleci dışarıdayken kontrolün üzerine giriş yaptığında çalışır.

ORTAK OLAYLAR

MouseHover

Fare kontrolün üzerinde belirli bir süre hareketsiz kalınca çalışır.

MouseLeave

Fare kontrolün üzerinden dışarı çıktığında çalışır.

MouseMove

Fare kontrolün üzerinde hareket ettiğinde çalışır.

MouseUp

Kontrolün üzerindeyken farenin basılı olan tuşu serbest bırakıldığında çalışır.

Resize

Kontrolün boyutları değiştirildiğinde çalışır.

ORTAK OLAYLAR

TextChanged

Kontrolün üzerindeki metin değiştirildiğinde çalışır.

VisibleChanged

Kontrolün Visible özelliği değiştirildiğinde çalışır.

ORTAK METOTLAR

Focus

İmleç kontrolün üzerine getirilerek o kontrol aktif haline getirilir.

Hide

Kontrol gizlenir.

Show

Gizlenmiş olan kontrol gösterilir.

FORM ÖZELLİKLERİ

AcceptButton

Form üzerindeyken ENTER tuşuna basıldığında, sanki burada belirtilen butona fare ile tıklanmış gibi işlem yapılır.

CancelButton

Form üzerindeyken ESC tuşuna basıldığında, sanki burada belirtilen butona fare ile tıklanmış gibi işlem yapılır.

ControlBox

False olarak ayarlanırsa formun başlığında en solda duran simge görünmez olur.

FormBorderStyle

Formun kenarlık stili. Formun yeniden boyutlandırılabilir olması gibi özellikler buradan ayarlanır.

FORM ÖZELLİKLERİ

Icon

Formun başlığının en solundaki simge buradan seçilir.

MaximizeBox

False olarak ayarlanırsa formu tam ekran yapma butonu görünmez.

MinimizeBox

False olarak ayarlanırsa formu simge durumuna küçültme butonu görünmez.

Opacity

Formun şeffaflığı

StartPosition

Formun başlangıç pozisyonu.

FORM OLAYLARI

Activated

Form aktif pencere olduğunda (üzerine tıklandığında, ön plana getirildiğinde) çalışır.

FormClosing

Form kapatılırken (formun kapatılması için komut verildiğinde, kapatılmadan hemen önce) çalışır.

Load

Form oluşturulup hafızaya yüklendiğinde çalışır.

FORM METOTLARI

Close

Formu kapatır.

ÖNEMLİ

Form metotları çağrılırken form adı yazılmaz, sadece metodun adı yazılır.

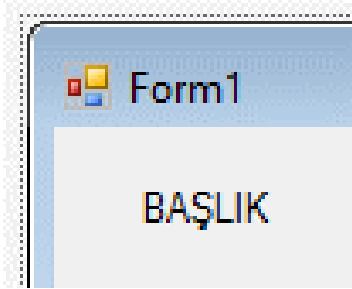
```
anaForm.Close(); // YANLIŞ  
Close(); // DOĞRU
```

Form adı yerine this yazılabilir. Ancak gereksizdir, bu yüzden tavsiye edilmez.

```
this.Close(); // DOĞRU AMA GEREKSİZ
```

LABEL

Formun üzerine kullanıcının değiştiremeyeceği bir yazı yerleştirmek için kullanılır.



LABEL ÖZELLİKLERİ

Text

Kontrolün üzerinde görünen metin. Kullanıcı tarafından değiştirilemez.

TEXTBOX

Kullanıcının metin girişı yapabilmesi için kullanılır.

A screenshot of a web browser showing a text input field. The field is rectangular with a thin border and contains the text "SAKARYA" in a blue, sans-serif font. The text box is set against a light gray background.

TEXTBOX ÖZELLİKLERİ

MaxLength

Metin kutusuna girilebilecek maksimum karakter sayısı

MultiLine

True olarak ayarlanırsa metin kutusuna birden çok satır girilebilir

PasswordChar

Boş bırakılırsa girilen metin aynen görünür. Bu özelliğe örneğin * gibi bir karakter atanırsa, metin girildiğinde girilen karakterlerin yerine hep atanan o karakter görünür. Genellikle şifre girişi için kullanılır.

ReadOnly

True olarak ayarlanırsa, girilen metin seçilebilir ve kopyalanabilir ancak değiştirilemez.

TEXTBOX ÖZELLİKLERİ

ScrollBars

MultiLine özelliğinin True olması durumunda, yatay ve dikey kaydırma çubuklarının görünür olup olmayacağı bu özellikten ayarlanır.

TEXTBOX OLAYLARI

GotFocus

Metin kutusu seçili halde geldiğinde (fare ile tıklandığında vs.) çalışır.

LostFocus

Metin kutusu seçili olmaktan çıktığında çalışır.

TextChanged

Metin kutusundaki metin değiştirildiğinde çalışır.

TEXTBOX METOTLARI

AppendText

Metin kutusundaki metnin sonuna başka bir metin eklemek için kullanılır.

Clear

Metin kutusundaki tüm metni siler.

Copy

Metin kutusundaki metni panoya kopyalar

Cut

Metin kutusundaki metni panoya kopyalar ve metin kutusundan siler.

Paste

Panodaki metni metin kutusuna yapıştırır.

TEXTBOX METOTLARI

Undo

Son yapılan işlemi geri alır. (Metni bir önceki hale döndürür)

BUTTON

Bir işlemi başlatmak için kullanılır.



ÇIKIŞ

BUTTON ÖZELLİKLERİ

Image

Butonun üzerine yerleştirilecek resim bu özellikten ayarlanır.

BUTTON METOTLARI

PerformClick

Butonun üzerine tıklanmış gibi işlem yapar.

UYGULAMA 1

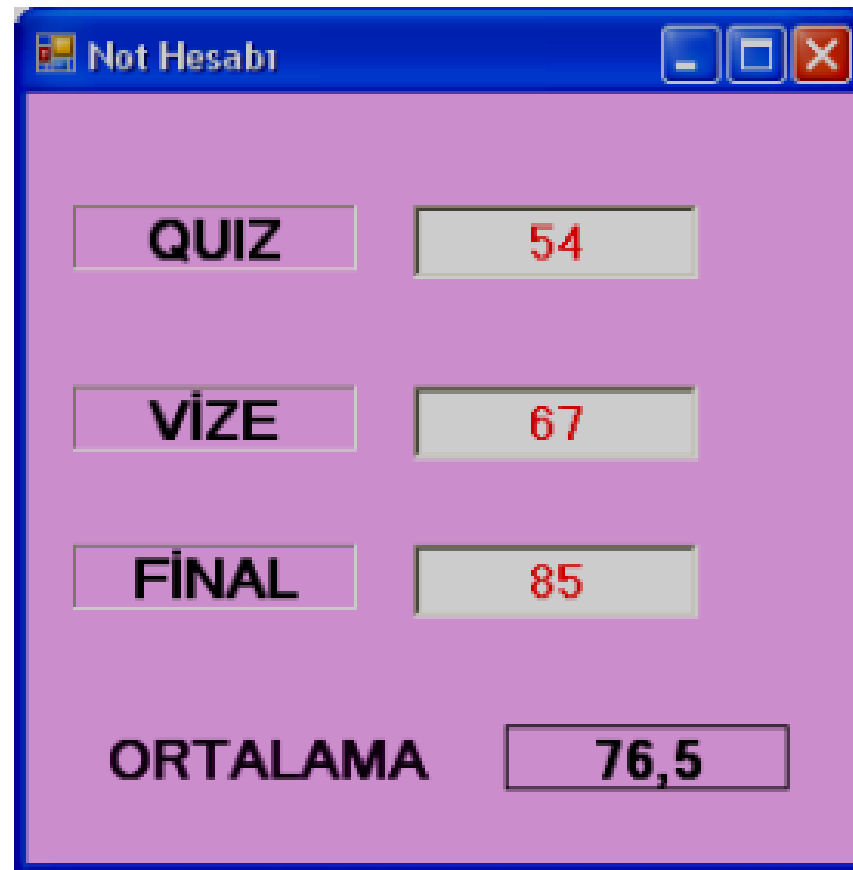
Yarıçapı girilen bir dairenin çevresinin ve alanını hesaplayan bir program yazınız.



Dairenin Çevresi ve Alanı	
Yarıçap	10
Dairenin Çevresi	62,8
Dairenin Alanı	314
Hesapla	Son

UYGULAMA 2

Girilen quiz, vize ve final notlarından ortalamayı hesaplayan bir program yazınız. Ortalama notu quiz'in %10'u, vizenin %30'u ve finalin %60'ı alınarak hesaplanacaktır.



The screenshot shows a Windows application window titled "Not Hesabı". The window has a blue title bar with standard minimize, maximize, and close buttons. The main area has a light purple background. It contains four rows of input fields. The first three rows are for "QUIZ", "VİZE", and "FİNAL", each with a corresponding score entered in red text: 54, 67, and 85 respectively. The fourth row is for "ORTALAMA" (Average), with the calculated value "76,5" displayed in black text.

Not Türü	Not Değeri
QUIZ	54
VİZE	67
FİNAL	85
ORTALAMA	76,5

CHECKBOX

Birden fazla seçenek arasından birini ve birkaçını kullanıcıya seçtirmek için kullanılır.

Aynı anda birden fazla seçenek seçilebilir.

- ☐ Resim
- ☒ Müzik
- ☒ Sinema
- ☒ Tiyatro

CHECKBOX ÖZELLİKLERİ

Checked

True olarak ayarlanırsa CheckBox seçili olur.

CheckedState

İşaretlenme durumu bu özellikten de ayarlanabilir.

Checked: İşaretli

Unchecked: İşaretsiz

Indeterminate: Belirsiz. Bu durum genellikle birden fazla eleman seçildiğinde bazıları işaretli bazıları işaretsizse kullanılır.

Öznitelikler:

☐ Salt okunur (Yalnızca klasördeki dosyalara uygulanır)

☒ Gizli

Gelişmiş...

CHECKBOX OLAYLARI

CheckedChanged

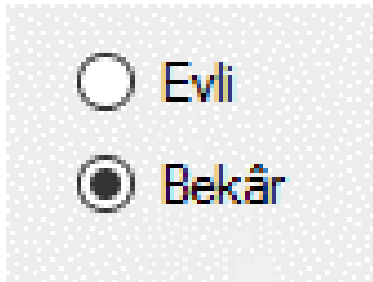
Seçili olma durumu değiştirildiğinde çalışır.

RADIOBUTTON

Birden fazla seçenek arasından birini kullanıcıya seçtirmek için kullanılır.

Aynı grubun içindeki radyo düğmelerinde aynı anda birden fazla seçenek seçili olamaz.

Farklı grumlardaki radyo düğmeleri birbirinden bağımsız çalışır.



RADIOBUTTON ÖZELLİKLERİ

Checked

True olarak ayarlanırsa CheckBox seçili olur.

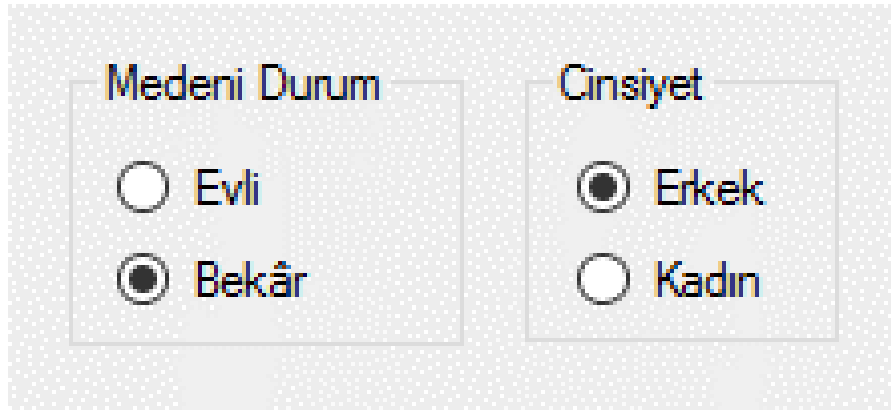
RADIOBUTTON OLAYLARI

CheckedChanged

Seçili olma durumu değiştirildiğinde çalışır.

GROUPBOX

Kontrolleri gruplamak için kullanılır. Başlığı vardır. (Text özelliğinden değiştirilir.)



The image shows a screenshot of a Windows application window with a light gray background. Inside the window, there are two GroupBox controls. The first GroupBox is titled "Medeni Durum" and contains two radio buttons: "Evli" (unselected) and "Bekâr" (selected). The second GroupBox is titled "Cinsiyet" and contains two radio buttons: "Erkek" (selected) and "Kadın" (unselected).

PANEL

Kontrolleri gruplamak için kullanılır. Başlığı yoktur.

The image shows a web form with two distinct groups of radio buttons, each enclosed in a rectangular box. The first box on the left contains two options: 'Evli' (unselected) and 'Bekâr' (selected). The second box on the right contains two options: 'Erkek' (selected) and 'Kadın' (unselected). The background of the form area has a light gray dotted pattern.

<input type="radio"/> Evli	<input checked="" type="radio"/> Erkek
<input checked="" type="radio"/> Bekâr	<input type="radio"/> Kadın

UYGULAMA 3

Ekrandan girilecek iki sayının üzerinde toplama, çıkarma, çarpma, bölme ve üs alma işlemlerini yapan basit bir hesap makinesi yapınız.

The screenshot shows a Windows-style application window titled "Hesap Makinesi". The window has a light pink background. On the left side, there are two red buttons labeled "1. SAYI" and "2. SAYI". Below "1. SAYI" is a text box containing the number "12". Below "2. SAYI" is a text box containing the number "3". In the center, there is a red-bordered box titled "İşlem Tipi" containing five radio button options: "Toplama", "Çıkarma", "Çarpma" (which is selected), "Bölme", and "Üs Alma". To the right of this box is a blue button labeled "SONUÇ" and a text box below it containing the number "36". At the bottom of the window, there are two green buttons: "Hesapla" and "Çıkış".