

WINDOWS FORM KONTROLLERİ II



İÇİNDEKİLER

- Gelişmiş Kontroller
 - PictureBox kontrolü
 - WebBrowser kontrolü
 - Timer kontrolü
 - ProgressBar kontrolü
 - GroupBox kontrolü
 - ImageList kontrolü
 - DateTimePicker kontrolü
 - MonthCalendar kontrolü
 - RichTextBox kontrolü



HEDEFLER

- Bu üniteyi çalıştıktan sonra;
 - Gelişmiş kontrol nesnelerini tanıyabilecek,
 - Gelişmiş kontrol nesnelerinin özelliklerini düzenleyebilecek,
 - Gelişmiş kontrol nesneleriyle uygulamaları zenginleştirebilecek,
 - Kontrollerin özel ve genel amaçlarına uygun tasarımlar gerçekleştirebileceksiniz.



Atatürk Üniversitesi
Açıköğretim Fakültesi

**GÖRSEL
PROGRAMLAMA I**
Öğr. Gör.
Daha ORHAN

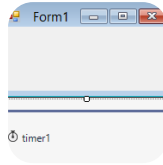
**ÜNİTE
9**



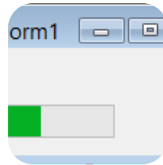
PictureBox



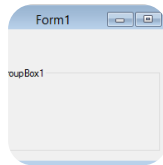
WebBrowser



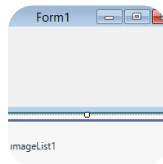
Timer



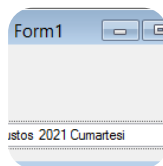
ProgressBar



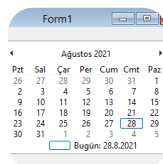
GroupBox



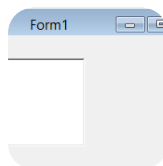
ImageList



DateTimePicker



MonthCalendar



RichTextBox

GİRİŞ

Microsoft Visual C# programlama dili, programcılarının Microsoft .Net Framework platformunu kullanmalarına imkan sağlayarak basit bir şekilde güçlü yazılımlar geliştirmelerine yardımcı olmaktadır. Visual Studio 2019 programı tarafından sağlanan yazılım geliştirme ortamı, birçok yeni sihirbaz ve önceki sürümlere istinaden yapılan iyileştirmeler sayesinde programcılarının üretkenliğine önemli derecede katkı sağlamaktadır.

Visual Studio 2019 programında yazılım geliştiricilerinin kullanabileceği Visual Basic, Python, C# gibi farklı programlama dilleri bulunmaktadır. Bu ünitedeki C# programlama diline yönelik bilgiler aktarılmaktadır. C# programlama dilinin tarihçesi uzun ve detaylı olmasına rağmen kısa bir bilgilendirme yapmanın faydalı olacağı düşünülmektedir. .Net Framework platformu Microsoft tarafından üretilmiştir. C ve C++ dillerinden türetilen C# programlama dili de bu platformda kullanılmak üzere tasarlanmış olan resmi dildir.

Görsel programlama dilleri geliştirilen uygulamaların belli bir kısmının grafiksel öğeler yardımıyla tasarlanmasına imkan vermektedir. Aynı zamanda formlar üzerine kontrol nesneleri yerleştirmeye de olanak sağlamaktadır. Kontrollerle ilgili işlemler, .Net Framework platformundaki System.Windows.Controls isim alanında bulunan sınıflarla gerçekleştirilmektedir.

Visual Studio 2019 programında C# programlama dili ile Windows Forms Uygulamalarında kullanılmak üzere geliştirilmiş birçok kontrol nesnesi araç kutusunda yer almaktadır. Bu kontrol nesnelerinin birbirine benzer özellikleri bulunsa da her bir kontrol ayrı bir amaç için geliştirilmiştir. Çok sayıda bulunan bu kontrolleri tek seferde anlatmak zor olduğu için bir önceki ünite bunlar içinden 9 tanesine yer verilmiştir. Bu ünite de farklı etkileşimlere sahip olan PictureBox, WebBrowser, Timer, ProgressBar, GroupBox, ImageList, DateTimePicker, MonthCalendar ve RichTextBox kontrolleri anlatılacaktır.



PictureBox kontrolü ile form üzerine resimler ekleyebilir ve bu resimlerin biçim özelliklerini ayarlayabiliriz.

GELİŞMİŞ KONTROLLER

Windows tabanlı uygulamalar geliştirilirken programcının ihtiyaç duyduğu resim ekleme alanları, zamanlayıcı, tarih bilgisi gibi standart kontrol nesnelere göre biraz daha gelişmiş kontroller söz konusu olabilir. Geliştirilen uygulamaların tamamında gerekli olmasa bile projemizin niteliğine bağlı olarak bu kontrollerden bir veya daha fazlası istenildiği zaman formumuza eklenebilmektedir.

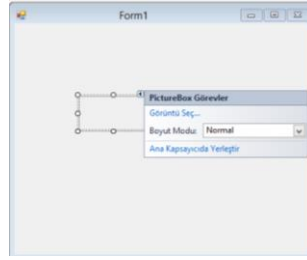
PictureBox Kontrolü

Uygulamamızın tasarımını gerçekleştirirken resimler ekleyerek daha canlı bir arayüz oluşturmak isteyebiliriz. Böyle bir durumda araç kutusunda bulunan PictureBox kontrolünü kullanmak işimize yarayacaktır. Bu kontrol sayesinde form üzerine resimler ekleyebilir ve bu resimlerin biçim özelliklerini ayarlayabiliriz.

PictureBox kontrolünü formumuz üzerine sürükleyip bırakmak yerine, araç kutusundan seçtikten sonra formda istediğimiz alana fare yardımıyla tarayarak

boyutlandırmak resimleri için ayırdığımız bölgeyi ayarlamak için daha elverişli bir yöntemdir. Ayrıca kontrolün form üzerindeki konumunu ayarlamak için özellikler panelinde bulunan *location* seçeneğini kullanabiliriz. Bu seçenekte bulunan ilk sayı kontrolün formun sağ kısmına göre piksel cinsinden uzaklığını, ikinci sayı ise formun üst kısmına göre piksel cinsinden uzaklığını ifade etmektedir. Bu yöntemlerin haricinde PictureBox kontrolünü form üzerinde konumlandırmak için özellikler panelinde bulunan *Dock* seçeneğindeki 6 farklı yöntemden birini tercih etmemiz de mümkündür.

Kontrolümüzü form üzerine yerleştirdikten sonra Şekil 9.1.'de gösterildiği gibi sağ üst kısmında bulunan ok'a tıklayıp *Görüntü Seç...* seçeneğini işaretleyerek *Kaynak Seç* penceresini açmamız gerekmektedir. Bu pencereyi özellikler panelinde bulunan *Image* seçeneğine tıklayarak da açabiliriz.

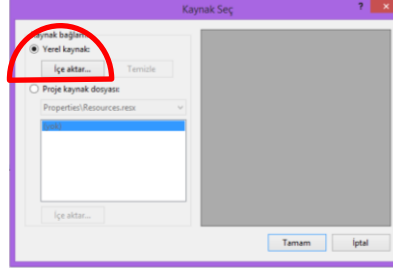


Şekil 9.1. PictureBox Kontrolü



Kaynak Seç penceresinde, resim dosyamızın hangi kaynaktan alınacağını belirlememiz gerekmektedir.

Açılan Kaynak Seç penceresinde öncelikle resim dosyamızın hangi kaynaktan alınacağını belirlememiz gerekmektedir. Kullanıcının görmesini istediğimiz resim dosyası uygulamanın geliştirildiği veya uygulamanın kurulacağı bilgisayarda ise Şekil 9.2.'de gösterildiği gibi *Yerel Kaynak* seçeneğini işaretlememiz gerekmektedir.

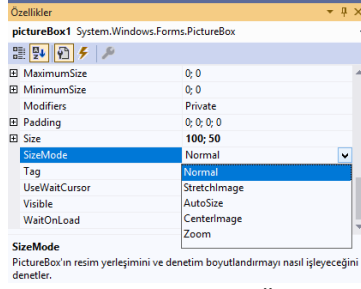


Şekil 9.2. Kaynak Seç Penceresi

Yerel kaynak seçeneği işaretlenip *İçerik aktar...* butonu tıklanırsa ekrana *Aç* penceresi gelir. Bu pencereden formumuza eklemek istediğimiz resmin dosya yolu belirtilerek veya resmin bulunduğu konum üzerinden seçim işlemi gerçekleştirilerek *Aç* butonuna tıklanır. Bu işlemin ardından seçilen resme ait önizleme kaynak seç penceresinin sağ tarafında görüntülenecektir. Bu pencerede *Temizle* butonuna tıklanırsa eklenen resim silinmiş olur. *Tamam* butonuna tıklanırsa da seçilen resme ait dosya bilgisi ilgili PictureBox kontrolünün *Image* özelliğine aktarılır. Bu işlem tamamlandıktan sonra form üzerinde PictureBox ile ayrılan alan içinde seçilen resim görüntülenecektir.

Yüklemek istediğimiz resmin çözünürlük değeri yüksek ise kontrolümüz üzerinde belli bir kısmı görüntülenir. Resmi orijinal boyutta görüntülemek için

PictureBox kontrolünün boyutlarını değiştirebilirsiniz. Ancak bu durum tasarladığımız form yapısının bozulmasına neden olabilir. Bu yüzden resmi eklemiş olduğumuz kontrolün boyutlarına göre düzenlemek daha elverişli olacaktır. Özellikler panelinde bulunan *SizeMode* seçeneğini kullanarak eklediğimiz resmi PictureBox kontrolünün boyutlarına göre düzenleyebiliriz. *SizeMode* özelliği *Normal*, *StretchImage*, *AutoSize*, *CenterImage* ve *Zoom* olmak üzere 5 farklı seçenek sağlamaktadır. PictureBox kontrolüne ait *SizeMode* özelliğinin ekran görüntüsü Şekil 9.3.'de gösterilmektedir.



Şekil 9.3. Sizemode Özelliği



SizeMode özelliği
Normal, StretchImage, AutoSize, CenterImage ve Zoom olmak üzere 5 farklı görünüm alternatifi sunmaktadır.

Normal: Eklenen resim PictureBox kontrolünün içine sol üst köşeden itibaren konumlanarak görüntülenir. Bu seçenek aktif olduğu zaman iki farklı durum oluşabilir. Bunlardan biri resmin boyutlarının kontrolün boyutlarından büyük olduğu durumda resmin bir kısmının görüntülenmesidir. Diğer bir durum ise resmin boyutlarının kontrolün boyutlarından küçük olmasıdır. Bu durumda da PictureBox kontrolünün resim ile doldurulamayan alanlarında kontrolün arka plan rengiyle doldurularak görüntülenir.

StretchImage: Eklenen resmin boyutu PictureBox kontrolünün boyutuna göre ayarlanarak görüntülenir. Resmin tamamı kontrolün boyutlarında görüntüleneceği için orijinal resim değişikliğe uğramış olacaktır. Bu durumda da çözünürlüğün bozulması gibi bir sonuçla karşılaşılabilir.

AutoSize: StretchImage özelliğinin tam tersi olacak şekilde çalışmaktadır. PictureBox kontrolünün boyutları, orijinal resim boyutlarına göre düzenlenerek form üzerinde görüntülenmesi sağlanır.

CenterImage: Resim PictureBox kontrolünün orta noktasına konumlanarak görüntülenir. Normal seçeneğinde olduğu gibi; resim kontrolden büyük olursa sığmayan kısımlar görüntülenmez, resim kontrolden küçük olursa doldurulamayan alanların kontrolün arka plan rengi ile doldurulur. CenterImage seçeneği ile Normal seçeneği arasındaki fark resmin konumlandığı noktadır. Normal seçeneğinde resim Kontrolün sol üst köşesinden itibaren konumlanır.

Zoom: Eklenecek olan resim genişletilerek veya daraltılarak PictureBox kontrolünün içinde görüntülenir. Genişletme ve daraltma işlemi, resmin orijinal boyutlarındaki en boy oranı korunarak gerçekleştirilir.

PictureBox kontrolüne resim ekleme işlemi yapılırken Kaynak Seç penceresinde iki farklı seçenek olduğunu belirtmiştik. Bu penceredeki *Kaynak Bağlamı* alanından *Yerel Kaynak* seçeneği ile resim ekleme işlemi gerçekleştirdik.

Bu yöntemle resim ekleyerek oluşturduğumuz programları başka bilgisayara taşıdığımızda söz konusu olan resim dosyalarını da taşımamız gerekmektedir. Bunun yerine resim dosyalarının C# projesindeki *Resx* uzantılı kaynak dosyasının içinde saklanmasını sağlayabiliriz. Kaynak seç penceresinde bulunan *Kaynak Bağlamı* alanındaki *Proje kaynak dosyası* seçeneği işaretlenmelidir. Bu işlemin ardından içe aktar butonunu kullanarak PictureBox kontrolünde görüntülemek istediğimiz resim dosyasını seçip işlemimizi tamamlayabiliriz.

Resimlerimizi ekleyip uygulamayı çalıştırdığımızda, PictureBox kontrolünün farklı metotlarını kullanarak resim üzerinde bir takım işlemler gerçekleştirebiliriz. Orijinal hali yatay veya dikey olan bir resmi döndürme ihtiyacı olabilir. Böyle bir durumda *RotateFlip()* metodunu kullanmamız gerekmektedir. *Rotate90FlipNone* seçeneğini *RotateFlip()* metoduna parametre olarak verip eklenen resmin saat yönünde 90 derece döndürülmesini sağlayabiliriz.

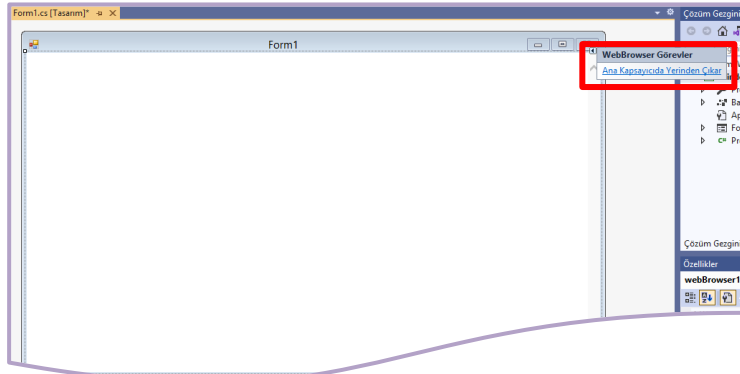
WebBrowser Kontrolü



WebBrowser kontrolü sayesinde formumuzda belirlenen alan içerisinde gitmesini istediğimiz adrese ait web sayfasını görüntüleyebiliriz.

Uygulamalarımızda bazı durumlar için internet sayfalarının görüntülenmesi isteyebiliriz. Böyle bir durumda araç kutusunda bulunan WebBrowser kontrolünü formumuza eklememiz gerekmektedir. Bu kontrolün davranış biçimi bir tarayıcınınkinden farklı değildir. WebBrowser kontrolü sayesinde formumuzda belirlenen alan içerisinde gitmesini istediğimiz adrese ait sayfayı görüntüleyebiliriz.

WebBrowser kontrolünü eklediğimiz zaman form üzerinde bulunan alanın tümü kontrol tarafından kullanılmaktadır. Uygulamamıza başka kontroller de ekleyebilmek için bu alanı daraltmamız gerekebilir. Şekil 9.4.'de gösterildiği gibi kontrolün sağ üst köşesinde bulunan butona tıklayıp *Ana Kapsayıcıda Yerinden Çıkar* seçeneğini işaretleyerek form üzerinde yer açabiliriz. Özellikler panelinde bulunan *Dock* seçeneğinde bulunan alternatiflerde kullanılabilir.



Şekil 9.4. Webbrowser Kontrolü

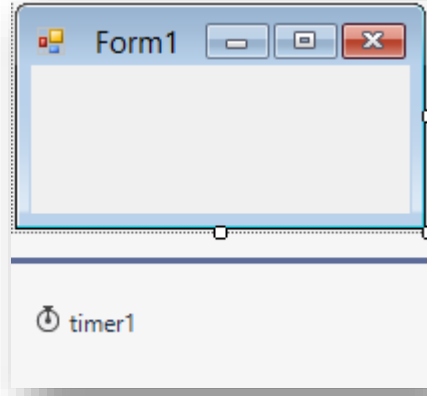
Görüntülenmek istenen web sayfasının adresi, kontrolün *Navigate* özelliğine yazılmalıdır. Bu adres bilgisini aldığımız alanda, "http://" başlığının yazılıp yazılmadığını kontrol etmemiz gerekmektedir.

Timer Kontrolü

Timer kontrolü zamanla yapılması gereken işlemlerin olduğu veya ekranda zaman bilgisinin bulunması gibi durumlarda kullanılmaktadır. Kontrolde zamanın ilerleme durumu milisaniye cinsinden hesaplanır. Uygulamamıza Timer kontrolünü eklemek istediğimiz zaman, kontrol form üzerinde görüntülenmez. Şekil 9.5.'de gösterildiği gibi Timer kontrolünün simgesi tasarım ekranındayken uygulamanın arka planında çalışmak üzere formun alt tarafında açılan bir panel üzerine konumlanır. Bu panel üzerindeki simgeye çift tıklayarak kod editöründe kontrolün en sık kullanılan *Tick* olayı açılır. Bu olay içerisinde *interval* özelliğine bağlı olarak zamanlanan olaylar gerçekleştirilir.



Timer kontrolünün en çok kullanılan özellikleri, Interval ve Enabled özellikleridir.



Şekil 9.5. Timer Kontrolü

System.Windows.Forms.Timer sınıfına ait olan Timer kontrolü ile .Net içerisinde gerçekleştirilebilecek üç tür zamanlama işlemi bulunmaktadır. Bunlar;

- System.Windows.Forms.Timer
- System.Timers.Timer
- System.Threading.Timer

Interval ve *Enabled* özellikleri timer kontrolünün en çok kullanılan özelliklerindendir. Interval özelliği, zaman aralığını milisaniye cinsinden belirtir. Bu özellik en fazla 2,147,483,647 değerini alabilir. Enabled özelliği ise kontrolün aktif veya pasif olmasını sağlamaktadır. Bu özellik *True* olduğu zaman kontrol aktif olarak çalışacaktır. *False* değerini atadığımızda da kontrol çalışmayacaktır. Ayrıca çalışma anında timer kontrolünü başlatmak için *Start*, durdurmak için *Stop* metotlarını kullanabiliriz.

Timer kontrolü birçok amaçla kullanılabilir. Örneğin formumuz üzerinde sistem saatini görüntülemek isteyebiliriz. Bunun için Now özelliğinin kullanılması gerekmektedir. Now özelliği kullanılarak sistem saatinin TextBox kontrolü üzerinde gösterimine yönelik örnek ProgressBar kontrolü anlatıldıktan sonra gösterilecektir.

ProgressBar Kontrolü

Windows formlarla çalışırken önemli kontrollerden biri olan ProgressBar, bir işlemin gerçekleştirilme esnasında ilerleme durumu hakkında bilgi vermeyi amaçlamaktadır. Bu ilerleme durumu bazen bir programın kurulumunu bazen de bir web sayfasının yüklenmesini gösterebilir. Projemizin özelliklerine bağlı olarak bu kontrolü ilişkilendireceğimiz seçenekler farklılaşabilir.

Özellikler panelinde bulunan **Value** değeri ProgressBar kontrolünün ne kadarlık bir kısmının boyalı olduğunu orantısal olarak ifade etmek için kullanılır. Bu özelliğin alabileceği değer aralığı yine özellikler panelinde bulunan **Maximum** ve **Minimum** özellikleriyle belirlenebilir. Varsayılan değer olarak 0 ile 100 arasında olacak şekilde ayarlanmıştır. ProgressBar kontrolünün minimum değerden maximum değere kadar hangi adım miktarıyla ilerleneceğini **Step** özelliği ile ayarlayabiliriz.



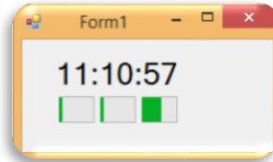
ProgressBar kontrolünün minimum değerden maximum değere kadar hangi adım miktarıyla ilerleyeceğini Step özelliği ile ayarlayabiliriz.

ProgressBar kontrolünü tek başına kullanmak işlevsel bir yöntem değildir. Genellikle **Timer** veya **BackgroundWorker** gibi kontrol nesneleriyle birlikte kullanılırlar. ProgressBar kontrolünü daha iyi kavrayabilmek için Timer kontrolüyle beraber kullanımına yönelik aşağıdaki örneğimizi inceleyelim.



Örnek

- ProgressBar kontrolünü kullanarak bir saatin saat, dakika ve saniye bileşenleri için görsel bir tasarım oluşturalım. Form üzerine bir adet label, üç adet progressbar ve bir adet timer kontrolü yerleştirelim. Uygulamanın kodları aşağıda, ekran görüntüsü de Şekil 9.6.'da verilmiştir.



Şekil 9. 6. Progressbar ve Timer Kontrollerinin Kullanımı

```
private void Form1_Load(object sender, EventArgs e)
{
    timer1.Start();
}

private void timer1_Tick(object sender, EventArgs e)
{
    DateTime saat = DateTime.Now;
    int saat_p = saat.Hour;
    int dakika_p = saat.Minute;
    int saniye_p = saat.Second;
    label1.Text = saat.ToString("HH:mm:ss");
    progressBar1.Value = saat_p;
    progressBar2.Value = dakika_p;
    progressBar3.Value = saniye_p;
}
```


Uygulamada label kontrolünün text özelliğinde sistem saatinden alınan saat dilimi gösterilmektedir. Progressbar kontrolleri ise saat dilimine ait her bir bileşenin ilerleme durumunu görüntülemektedir. Aynı zamanda uygulamayı çalıştırınca saatin aktif olarak başlaması için formun Load (yüklenme) olayına Timer kontrolüne start komutu verilmiştir.

GroupBox Kontrolü

GroupBox kontrolü, uygulamalarımızda form üzerine yerleştirilen kontrolleri belirli gruplar halinde görüntülemek için kullanılmaktadır. Tasarım ekranında ilgili kontrolleri bir bütün olarak yansıtip ön plana çıkarabileceği gibi genellikle birbirinden etkilenen RadioButton gibi kontrollerin de ayrılmasına olanak sağlamaktadır.

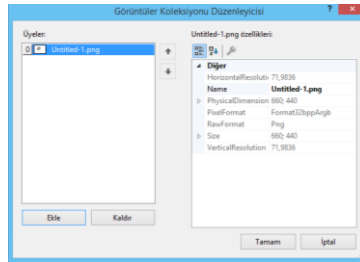
Form üzerine yerleştirilen GroupBox kontrolü, başlık yazısı ve çerçeveye sahiptir. Başlık yazısı ile grup içinde bulunan kontroller hakkın bilgi verilebilir. Özellikler panelinde bulunan Text seçeneği ile kontrole ait başlık yazısını düzenleme imkanına sahibiz.

Form üzerinde bulunan kontrolleri daha önceden yerleştirmiş olduğumuz GroupBox kontrolünün içine sürükleyip bırak yöntemi ile konumlandırabiliriz. Bununla beraber kontrol nesneleri daha önceden eklenmiş ise araç kutusundan GroupBox kontrolünü seçtikten sonra gruplanmak istenen kontrolleri fare ile çerçeve içine alarak gruplama işlemini gerçekleştirebiliriz.

ImageList Kontrolü

ImageList kontrolü projemizde kullanacağımız resim dosyalarının toplu olarak bir alan üzerinde tutulmasını ve tutulan bu resimlere diğer kontroller tarafından erişilmesini sağlamak amacıyla kullanılmaktadır. Araç kutusu üzerinde kontrole çift tıkladığımız zaman kontrol formumuz üzerinde görüntülenmeyecektir. ImageList kontrolüne ait simge, kullanıcı arabiriminin altında bulunan panel üzerine yerleşecektir.

ImageList kontrolünün içerisine resim ekleme, kontrolde var olan resimleri silme ve düzenleme işlemlerini gerçekleştirebiliriz. Bu işlemler özellikler panelinde bulunan Image seçeneğinin yanındaki butona tıklanarak açılan **Görüntüler Koleksiyonu Düzenleyicisi** penceresinden gerçekleştirilebilir. Şekil 9.7.'de gösterilen bu pencerenin sol tarafında bulunan üyeler kısmında daha önceden eklenen resimler bulunmaktadır.



Şekil 9.7. Görüntüler Koleksiyonu Düzenleyicisi Penceresi

ImageList kontrolü projemizde kullanacağımız resim dosyalarının toplu olarak saklanması amacıyla kullanılmaktadır.

Bu listeyi çoğaltmak için **Ekle** butonunu kullanabiliriz. Ekle butonuna tıkladıktan sonra karşımıza gelen **Aç** penceresinden resimler seçilerek **Tamam** butonuna tıklanır. Görüntüler koleksiyonu düzenleyicisi penceresinde görüntülenen resimleri seçerek **Kaldır** butonu ile listedeki seçenekleri azaltma imkanına da sahibiz. Ayrıca eklenme sırasına göre listelenen resimlerin sırasını yukarı ve aşağı yönlü ok tuşları yardımıyla değiştirebiliriz. Görüntüler koleksiyonu düzenleyicisi penceresinin sol tarafındaki alanda ise seçili olan resme ait özellikler görüntülenmektedir. Görüntüler koleksiyonu düzenleyicisi penceresine ImageList kontrolünün sağ üst köşesinde bulunan oka tıklayıp Görüntüleri Seç seçeneğini işaretleyerek de ulaşabiliriz.

Listesi oluşturulacak olan resimlerin genişlik ve yükseklik değerleri belirlenebilir. Bu işlem özellikler panelindeki **ImageSize** seçeneği ile gerçekleştirilmektedir. Bununla birlikte ImageList kontrolü içerisinde bulunan resimlerin renk derinliği **ColorDepth** özelliği kullanılarak ayarlanmaktadır.

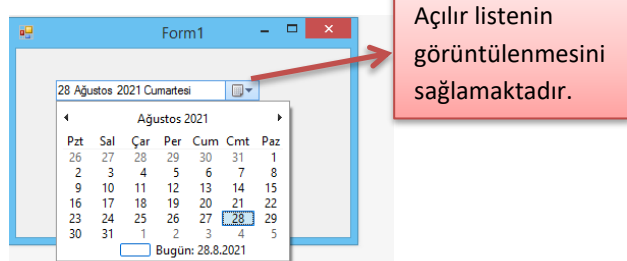
ImageList kontrolüne resimlerimizi ekledikten sonra herhangi bir kontrolün özellikler panelinde bulunan ImageList seçeneğini işaretleyerek oluşturduğumuz ImageList kontrolünü bağlamamız gerekmektedir. Bu işlemin ardında bağlantısını sağladığımız kontrolün ImageIndex özelliğinden index numarası aktarılan resmi ekleyebiliriz. Bu işlemin aynısını ImageKey özelliği ile resim dosyasına ait isim bilgisi ile de gerçekleştirebiliriz. Form üzerinde bulunan kontrol nesnelere ImageList kontrolünü bağlarken, bağlantısı yapılacak olan kontrolün bu özelliği destekleyip desteklemediği kontrol edilmelidir.



DateTimePicker kontrolü, tarih ve saat bilgisi girilirken kullanılmaktadır.

DateTimePicker Kontrolü

DateTimePicker kontrolü, tarih ve saat bilgisi girilirken kullanılmaktadır. Bu kontrol nesnesini formumuza ekleyip çalıştırdığımız zaman sistem tarihi başlangıç değeri olarak görüntülenir. Çalışma anında form üzerinde bulunan DateTimePicker kontrolünü fare ile seçtikten sonra ok tuşlarını kullanarak görüntülenen tarih bilgisi seçim olacak şekilde değiştirilebilmektedir. Bunun yanı sıra kontrolün yanında bulunan aşağı yönlü ok'a tıkladığımızda Şekil 9.8.'de gösterildiği gibi liste kutusunu açabiliriz. Bu alan üzerinden istenilen bir tarih seçilerek kontrol üzerinde görüntülenmesi sağlanabilir.



Şekil 9.8. Datetimepicker Kontrolü

Çalışma anında kontrolün görüntüleyebileceği tarih aralığı sınırlandırılabilir. Bu işlem için özellikler panelinde bulunan **MaxDate** ve **MinDate** seçeneklerini kullanabiliriz.

DateTimePicker kontrolünü formumuza eklediğimizde varsayılan değer olarak tarih bilgisi uzun tarih formatında gösterilmektedir. Bu durum özellikler panelinde bulunan **Format** seçeneği ile değiştirilebilir. Format özelliği bize dört farklı alternatif sunmaktadır. Bunlar;

- Long
- Short
- Time
- Custom

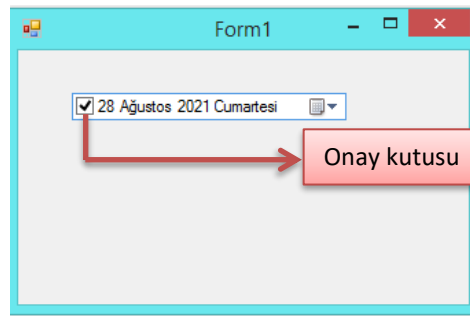
Bazı durumlarda DateTimePicker kontrolünün, çalışma anında bir liste kutusu yerine sayı kutusu gibi görünmesi istenebilir. Kontrolün böyle bir görüntüye sahip olması için özellikler panelinde bulunan **ShowUpDown** seçeneğine **True** değerini aktarmamız gerekmektedir. Böylelikle kontrol şekil değişikliğine uğramış olacaktır. Bu görünüm haliyle kontrole eklenen yukarı ve aşağı yönlü ok tuşlarını kullanarak tarih bilgisini değiştirebiliriz.

Çalışma anındaki görünüm seçeneği sadece tasarımla alakalıdır. Kontrol nesnesinden seçmiş olduğumuz tarih bilgisi DateTimePicker kontrolünün **Value** özelliğinde saklanır. Kontrolün Text özelliği String tipinde bilgi saklarken, Value özelliği **DateTime** tipinde bilgi saklamaktadır. Value özelliğinin sakladığı değer ToString() metodu ile String veri tipine dönüştürülebilir.

Uygulamalarımızda DateTimePicker kontrolünün onaylanarak değişiklik yapılmasına izin vermek isteyebiliriz. Bu durumda özellikler panelinde bulunan **ShowCheckBox** seçeneğini True olarak ayarlamamız gerekmektedir. Böyle bir değişiklik yapıldıktan sonra çalışma anında Şekil 9.9.'da gösterildiği gibi kontrolün sol tarafında bir onay kutusu görüntülenecektir. Bu onay kutusu işaretlenirse tarih bilgisi değiştirilebilmektedir. Aksi takdirde kontrol nesnesi pasif konumda olacaktır. Kod editöründe bu onay kutusunun işaretli olup olmadığı **Checked** özelliği ile kontrol edilebilmektedir.



MonthCalendar kontrolü, önemli tarihleri veya zaman dilimlerini işaretleyebilmemizi sağlamaktadır.



Şekil 9.9. Showcheckbox = True

MonthCalendar Kontrolü

MonthCalendar kontrolü, önemli tarihleri veya zaman dilimlerini işaretleyebilmemizi sağlamaktadır. DateTimePicker kontrolüne benzer özellikleri bulunsa da çalışma anında açılır bir liste yerine tüm bir takvim halinde görüntülenmektedir. MonthCalendar kontrolü ve DateTimePicker kontrolü ile

çalışırken DateTime sınıfına hakim olmamız gerekmektedir. Aşağıda bu sınıfa ait sıklıkla kullanılan özellik ve metotlardan bahsedilecektir.

Sistem tarihini kullanmak için DateTime yapısının *Today* özelliğinden istifade edilebilir. Form üzerine yerleştirilen bir TextBox kontrolünde sistem tarih bilgisini almak için; *textboxadı.text = Datetime.today.tostring();* kodu yazılabilir. Burada Textbox kontrolü string türünde bilgi görüntüleyeceği için DateTime sınıfından gelen veri dönüştürülmüştür. Uygulamalarımızı geliştirirken kontrollerin kullanacağı veri tiplerine de dikkat etmemiz gerekmektedir.

Sistem saatine ait tarih bilgisiyse beraber saat bilgisini de kullanabilmek için DateTime yapısına ait *Now* özelliğini kullanmamız gerekir. Bu özelliğin benzer seçeneklerin DateTimePicker kontrolünde bahsetmiştik. Kontrol alanında özellikler panelinde yapılan işlemlerin tamamı bu sınıfı kullanırken kod editöründe de gerçekleştirilebilmektedir. Örneğin Now özelliğini kullanarak sadece zaman bilgisi görüntülemek istiyorsanız, *ToShortTimeString()* metodunu kullanmanız gerekir.

Herhangi bir tarih bilgisinde verilen değere göre haftanın gününü öğrenebiliriz. Bu işlemi gerçekleştirmek için DayOfWeek özelliğini kullanmamız gerekmektedir. Benzer şekilde yılın kaçınıcı günü olduğuna dair bilgiye sahip olmak içinde DayOfYear özelliğinden istifade edebiliriz.



DateTime sınıfının özellikleri kullanarak sistem tarihine ait gün, ay ve yıl bilgileri ayrı ayrı çekilebilir.

DateTime sınıfının özellikleri kullanılarak sistem tarihine ait gün, ay ve yıl bilgileri ayrı ayrı çekilebilir. *Day* özelliği gün bilgisini öğrenmek için kullanılabilir. Belirtilen tarihin ay bilgisini öğrenmek için *Month* özelliği, yıl bilgisine öğrenmek içinde *Year* özelliği kullanılmaktadır.

Bazı uygulamalarda istisnası bulunan durumların gözden kaçırılmaması gerekebilir. Örneğin bir yıla ait kaç gün bulunduğuna dair bir bilgi her zaman aynı sonucu vermeyebilir. Çünkü dört yılda bir olmak üzere bir yıldaki gün sayısı 366 olabilmektedir. Benzer şekilde bir yıl içinde bulunan her ay aynı gün sayısına sahip değildir. Gün sayısına bağlı olarak maaş bilgisinin hesaplandığı bir kurumda bu seçeneklerin dikkate alınması gerekir. Bu tarz uygulamalarda DateTime sınıfına ait *DaysInMonth()* ve *IsLeapYear()* metotları kullanılabilir.

DaysInMonth() metodu integer tipinde değer üretir. 2 parametre alacak şekilde kullanılmaktadır: *DateTime.DaysInMonth(YIL, AY);* Burada metoda ait parantezler içinde girilecek olan ilk parametrede, gün sayısını öğrenmek istediğimiz Ay'a ait Yıl bilgisi olmalıdır. Metodun ikinci parametresinde ise gün sayısını öğrenmek istediğimiz Ay bilgisi yazılmalıdır. Bu kodu programınıza yazıp parametre değerlerini sırasıyla 2020 ve 2 girdiğiniz zaman, sonuç olarak 29 değerini ürettiğini görebilirsiniz.

IsLeapYear() metodu ise biraz daha genel bir kullanıma sahiptir. Yani herhangi bir yıla ait ayların kaç gün geçtiğine dair bir bilgiyi elde edemeyiz. Sadece belirtilen yılın gün sayısı hakkında bir yorum yapmamızı ve dolaylı olarak şubat ayı hakkında fikir sahibi olmamızı sağlamaktadır. Boolean tipinde değer üreten bu metoda parametre olarak gönderilen yıl 366 gün ise True sonucunu ortaya çıkarır.

Kod editörüne yazacağımız `DateTime.IsleapYear(2021)`; bu satırdan sonra 2021 yılının 365 gün olduğuna dair bir bilgi kullanıcı ile paylaşılabilir.

Tarih bilgisi üzerinde işlem yapabildiğimiz gibi saat dilimleriyle alakalı işlemleri gerçekleştirebilmemiz mümkündür. `Date.Time` sınıfına ait `Hour`, `Minute` ve `Second` özelliklerini kullanarak zaman bilgisindeki saat, dakika ve saniye alanları üzerinde işlemler gerçekleştirebiliriz. Örneğin `DateTime` tipinde saklanan bir değişken üzerinde saat bilgisini çekip, bu saat bilgisindeki dakika bilgisini kullanıcıya mesaj yoluyla verebiliriz. Bu örneği işe başlangıç saatindeki gecikmenin hesaplandığı basit bir proje içinde düşünelim. İş yerine gelen her çalışanın form üzerinde bulunan buton kontrolüne tıklaması gerektiğini varsayalım. Böylelikle butonun tıklanma olayında dakika bilgisi gösteren mesajı yazarak kullanıcılar için bir uyarı oluşturulması sağlanabilir.

Saat dilimi içindeki bileşenleri ayrı ayrı çekerek işlemler yapılabildiği gibi, bu bileşenleri ayırmadan hazır metotları kullanarak da bir takım işlemler gerçekleştirilebilmektedir. Saat diliminde bulunan saat, dakika ve saniye bileşenlerine `AddHours()`, `AddMinutes()` ve `AddSecond()` metotlarını kullanarak ekleme ve çıkarma işlemi yapılabilir. Bu metotlar `DateTime` tipinde tanımlanan değişkenlerle kullanılmaktadır. Parametre olarak gönderilen değer negatif olduğu zaman eksiltme, pozitif olduğu zaman artırma işlemi gerçekleştirilir. Örneğin `DateTimeDegiskeni.AddHours(2)`; kod satırı yazıldığında, zaman bilgisini tutan değişkendeki saat bilgisi 2 saat artırılırken; `DateTimeDegiskeni.AddSecond(-40)`; kod satırı yazıldığında da, zaman bilgisini tutan değişkendeki saniye bilgisi 40 saniye azaltılmaktadır.



RichTextBox kontrolü ile kullanıcının metin girebilmesine izin verildiği gibi daha önceden hazırlanan bir metin de kullanıcıya gösterilebilmektedir.



Bireysel Etkinlik

- Form üzerine eklediğiniz bir textbox kontrolünde sistem saatini görüntüleyin. Ardından ekleyeceğiniz bir buton yardımıyla bu saat dilimindeki dakika bileşenini 20 dakika artıracak şekilde hazırlayın.

RichTextBox Kontrolü

Windows form uygulamalarında kullanıcının metin girişi yapabildiği `TextBox` kontrolü bazı durumlarda yetersiz kalabilmektedir. Örneğin biçimlendirme özellikleri `TextBox` kontrolünde kullanılamamaktadır. Kullanıcının veri girişi yaptığı alanlarda daha gelişmiş seçenekler sunmak için `RichTextBox` kontrolü kullanılabilir. Bu kontrol sayesinde yazıların font özellikleri ve rengi değiştirilebilmektedir. Aynı zamanda metnin farklı yerlerinde farklı seçeneklerin uygulanması da sağlanabilmektedir.

`RichTextBox` kontrolü ile kullanıcının metin girebilmesine izin verildiği gibi daha önceden hazırlanan bir metin de kullanıcıya gösterilebilmektedir. Hazır bir metni `RichTextBox` kontrolünün içerisine ekleyebilmek için özellikler alanında

bulunan **Text** seçeneğini kullanabiliriz. Aynı işlem kontrolün sağ köşesinde bulunan ok'a tıklanıp **Metin Satırlarını Düzenle...** seçeneğine tıklanarak da gerçekleştirilebilir. Açılan **Dize Koleksiyonu Düzenleyicisi** penceresine istenilen metinler yazılarak **Tamam** butonuna tıkla işlemi gerçekleştirilir. Böylelikle hazır bir metin çalışma anında kullanıcıya gösterilmiş olacaktır. Kullanıcı kendi yazdığı metinleri düzenleyebildiği gibi hazır sunulan bu metin üzerinde de işlemler gerçekleştirebilir. Eğer kullanıcının hazır sunulan metin üzerinde herhangi bir değişiklik yapmasını istemiyorsanız özellikler panelinde bulunan **ReadOnly** seçeneğini **True** olarak ayarlamamız gerekmektedir. Böylece kullanıcı RichTextBox üzerinde bulunan metni sadece okuyabilir. ReadOnly özelliği True olarak ayarlanan RichTextBox kontrolünde seçim işlemi yapıldıktan sonra Windows kısayol tuşu olan Ctrl+X işlem gerçekleştirmezken Ctrl+C kısayol tuşu çalışmaktadır. Böyle bir durumda metnin kesilemez ve aynı zamanda kopyalanamaz bir halde olmasını istersek özellikler panelinde bulunan **ShortcutsEnabled** özelliğini de **False** olarak ayarlamamız gerekir.

RichTextBox kontrolü, programcının sınırlandırmalar yapmasına veya kullanıcının bir takım kolaylıklardan istifade etmesine imkan sunan birçok özelliği bulunmaktadır. Özellikler panelinde bulunan bu seçeneklerden sık kullanılanlarını inceleyelim.

DetectUrls: Çalışma anında yazılan web adreslerinin otomatik olarak bağlantı metni şeklinde gösterilmesinin sağladığı özelliktir. **True** değeri aktifleştirildiğinde adres yazıldıktan sonra altı çizili mavi renk ile gösterilmektedir. Bu işlem sadece tasarım kısmında kullanılmaktadır. Bağlantının gerçekleştirilmesi için RichTextBox kontrolünün **LinkClicked** olayında, adresin herhangi bir tarayıcı programı ile açılmasına yönelik kodların yazılması gerekmektedir.

EnableAutoDragDrop: Sürükle bırak yönteminin RichTextBox içerisinde kullanılıp kullanılmayacağını ayarlamamızı sağlar.

ZoomFactor: RichTextBox kontrolünün içindeki metinlerin çalışma anındaki ölçeklendirme katsayısını ayarlamamızı sağlar. Normal görüntüleme değeri 1'dir.

Çalışma anında kullanıcı RichTextBox kontrolüne yazdığı metinleri kaydetmek isteyebilir. Böyle bir durumda **SaveFile()** metodu kullanılarak dosya içeriği RTF biçiminde bilgisayara kaydedilebilir. Kod editöründe bu işlemi yaptırmak istediğimiz alana **RichTextBoxAdı.SaveFile("DosyaYoluAdıUzantısı");** kodunu yazarak kaydetme işlemini gerçekleştirebiliriz. Dikkat ederseniz bu kod ile **SaveFile()** metodunun parametresini biz belirlemekteyiz. Bu işlemin kullanıcı tarafından gerçekleştirilmesi için **SaveFileDialog** kontrolünü projemize ekleyip açılmasını sağlamalıyız.

Savefile() metodu varsayılan olarak RTF biçimde dosya hazırlamaktadır. Dosya uzantısını **TXT** biçiminde ayarlamak için metodun isteğe bağlı olan ikinci parametresine **RichTextBoxStreamType.PlainText** yazmamız gerekir.



RichTextBox kontrolünün **Savefile()** metodu, varsayılan olarak RTF biçimde dosya hazırlamaktadır.



Özet

- Uygulamalarımızın tasarımını gerçekleştirirken resimler ekleyerek daha canlı bir arayüz oluşturmak isteyebiliriz. Böyle bir durumda araç kutusunda bulunan PictureBox kontrolünü kullanmak işimize yarayacaktır. Bu kontrol sayesinde form üzerine resimler ekleyebilir ve bu resimlerin biçim özelliklerini ayarlayabiliriz.
- PictureBox kontrolünü formumuz üzerine sürükleyip bırakmak yerine, araç kutusundan seçtikten sonra formda istediğimiz alana fare yardımıyla tarayarak boyutlandırmak resimleri için ayırdığımız bölgeyi ayarlamak için daha elverişli bir yöntemdir. Ayrıca kontrolün form üzerindeki konumunu ayarlamak için özellikler panelinde bulunan location seçeneğini kullanabiliriz. Bu seçenekte bulunan ilk sayı kontrolün formun sağ kısmına göre piksel cinsinden uzaklığını, ikinci sayı ise formun üst kısmına göre piksel cinsinden uzaklığını ifade etmektedir.
- Uygulamalarımızda bazı durumlar için internet sayfalarının görüntülenmesi isteyebiliriz. Böyle bir durumda araç kutusunda bulunan WebBrowser kontrolünü formumuza eklememiz gerekmektedir. Bu kontrolün davranış biçimi bir tarayıcınınkinden farklı değildir. WebBrowser kontrolü sayesinde formumuzda belirlenen alan içerisinde gitmesini istediğimiz adrese ait sayfayı görüntüleyebiliriz.
- Timer kontrolü zamanla yapılması gereken işlemlerin olduğu veya ekranda zaman bilgisinin bulunması gibi durumlarda kullanılmaktadır. Kontrolde zamanın ilerleme durumu milisaniye cinsinden hesaplanır. Uygulamamıza Timer kontrolünü eklemek istediğimiz zaman, kontrol form üzerinde görüntülenmez. Timer kontrolünün simgesi tasarım ekranındayken uygulamanın arkaplanında çalışmak üzere formun alt tarafında açılan bir panel üzerine konumlanır. Bu panel üzerindeki simgeye çift tıklayarak kod editöründe kontrolün en sık kullanılan Tick olayı açılır. Bu olay içerisinde interval özelliğine bağlı olarak zamanlanan olaylar gerçekleştirilir.
- Windows formlarla çalışırken önemli kontrollerden biri olan ProgresBar, bir işlemin gerçekleştirilme esnasında ilerleme durumu hakkında bilgi vermeyi amaçlamaktadır. Bu ilerleme durumu bazen bir programın kurulumunu bazen de bir web sayfasının yüklenmesini gösterebilir. Projemizin özelliklerine bağlı olarak bu kontrolü ilişkilendireceğimiz seçenekler farklılaşabilir.
- GroupBox kontrolü, uygulamalarımızda form üzerine yerleştirilen kontrolleri belirli gruplar halinde görüntülemek için kullanılmaktadır. Tasarım ekranında ilgili kontrolleri bir bütün olarak yansıtip ön plana çıkarabileceği gibi genellikle birbirinden etkilen RadioButton gibi kontrollerin de ayrılmasına olanak sağlamaktadır.
- ImageList kontrolü projemizde kullanacağımız resim dosyalarının toplu olarak bir alan üzerinde tutulmasını ve tutulan bu resimlere diğer kontroller tarafından erişilmesini sağlamak amacıyla kullanılmaktadır. Araç kutusu üzerinde kontrole çift tıkladığımız zaman kontrol formumuz üzerinde görüntülenmeyecektir. Şekilde gösterildiği gibi ImageList kontrolüne ait simge, kullanıcı arabiriminin altında bulunan panel üzerine yerleşecektir.
- DateTimePicker kontrolü, tarih ve saat bilgisi girilirken kullanılmaktadır. Bu kontrol nesnesini formumuza ekleyip çalıştırdığımız zaman sistem tarihi başlangıç değeri olarak görüntülenir. Çalışma anında form üzerinde bulunan DateTimePicker kontrolünü fare ile seçtikten sonra ok tuşlarını kullanarak görüntülenen tarih bilgisi seçim olacak şekilde değiştirilebilmektedir.
- MonthCalendar kontrolü, önemli tarihleri veya zaman dilimlerini işaretleyebilmemizi sağlamaktadır. DateTimePicker kontrolüne benzer özellikleri bulunsu da çalışma anında açılır bir liste yerine tüm bir takvim halinde görüntülenmektedir. MonthCalendar kontrolü ve DateTimePicker kontrolü ile çalışırken DateTime sınıfına hakim olmamız gerekmektedir.
- Kullanıcının veri girişi yaptığı alanlarda daha gelişmiş seçenekler sunmak için RichTextBox kontrolü kullanılabilir. Bu kontrol sayesinde yazıların font özellikleri ve rengi değiştirilebilmektedir. Aynı zamanda metnin farklı yerlerinde farklı seçeneklerin uygulanması da sağlanabilmektedir.

DEĞERLENDİRME SORULARI

1. Aşağıdakilerden hangisi PictureBox kontrolüne ait SizeMode özelliğinin resim görüntüleme seçeneklerinden biri değildir?
 - a) Normal
 - b) StretchImage
 - c) AutoSize
 - d) Padding
 - e) CenterImage
2. Uygulamalarımızda internet sayfalarını görüntüleyebilmek için kullanacağımız kontrol aşağıdakilerden hangisidir?
 - a) PictureBox
 - b) WebBrowser
 - c) GroupBox
 - d) RichTextBox
 - e) MonthCalendar
3. Timer kontrolünün interval özelliği hangi zaman aralığını ifade etmektedir?
 - a) Milisaniye
 - b) Saniye
 - c) Dakika
 - d) Saat
 - e) Gün
4. Aşağıdakilerden hangisi bir işlemin gerçekleştirilme esnasında ilerleme durumu hakkında bilgi vermesini sağlayan kontroldür?
 - a) PictureBox
 - b) GroupBox
 - c) ProgressBar
 - d) ImageList
 - e) DateTimePicker
5. Aşağıdakilerden kontrollerden hangisi form üzerine yerleştirilen kontrolleri belirli gruplar halinde görüntülemek için kullanılmaktadır?
 - a) PictureBox
 - b) WebBrowser
 - c) Timer
 - d) ProgressBar
 - e) GroupBox

6. ImageList kontrolü ile listesi oluşturulacak olan resimlerin genişlik ve yükseklik değerleri hangi özellik ile ayarlanabilmektedir?
 - a) ImageSize
 - b) ColorDepth
 - c) Text
 - d) ImageKey
 - e) ImageIndex

7. DateTimePicker kontrolünün çalışma anında görüntüleyebileceği tarih aralığı hangi özellikler kullanılarak belirlenebilir?
 - a) MaxDate – MinDate
 - b) Long – Short
 - c) Time – Costum
 - d) Maxvalue – MinValue
 - e) ShowUp – ShowDown

8. Aşağıdakilerden hangisi form üzerine yerleştirilen bir TextBox kontrolünde sistem tarih bilgisini almak için kullanılabilecek koddur?
 - a) DateTimeDegiskeni.AddSecond(-40);
 - b) textboxadı.text = Datetime.today.tostring();
 - c) ToShortTimeString()
 - d) DateTime.DaysInMonth(YIL, AY);
 - e) DateTime.IsleapYear(2021);

9. DateTime tipinde saklanan bir değişken üzerinde saat bilgisini çekmek için hangi özelliği kullanmamız gerekmektedir?
 - a) Minute
 - b) Second
 - c) Month
 - d) Year
 - e) Hour

10. Uygulama üzerinde bulunan RichTextBox kontrolündeki metin bilgisinin üzerinde değişiklik yapılmasını engellemek için hangi özellik kullanılmalıdır?
 - a) EnableAutoDragDrop
 - b) DetectUrls
 - c) ShortcutsEnabled
 - d) ReadOnly
 - e) ZoomFactor

Cevap Anahtarı

1.d, 2.b, 3.a, 4.c, 5.e, 6.a, 7.a, 8.b, 9.e, 10.d

YARARLANILAN KAYNAKLAR

- Aktaş, V. (2010). *Visual studio 2010 ile her yönüyle C# 4.0* (2. Baskı). İstanbul: Kodlab Yayın.
- Aktaş, V. (2020). *Visual studio 2019* (1.Baskı). İstanbul: 01 Yayınları.
- Atasever, V. (2017). *C# 7 Uygulamalarla C# programlama dilini keşfedin* (2. Baskı). Kocaeli: Level Kitap.
- Demirli, N. & İnan, Y. (2005). *Visual C# .Net* (3. Baskı). Ankara: Palme Yayıncılık.
- Sharp, J. (2011). *Adım adım Microsoft visual C# 2010*. (Çev. T. Buldu). Ankara: Arkadaş Yayınevi.
- Uzunköprü, S. (2017). *Projeler İle C# 7.0 ve SQL Server 2016* (7. Baskı). İstanbul: Kodlab Yayın.
- Yanık, M. (2008). *Visual studio 2008 ile Microsoft visual C# 3.0 for .Net framework 3.5* (1. Baskı). Ankara: Seçkin Yayıncılık.
- Yücedağ, M. (2020). *C# Eğitim kitabı* (3. Baskı). İstanbul: Dikeyeksen.