

YENİ BAŞLAYANLAR VE ÇOCUKLAR İÇİN

C# PROGRAMLAMA

MURAT YÜCEDAĞ

“Siz genç arkadaşlar, yorulmadan beni takibe ahdetmişsiniz. İşte ben bu sözden çok duygulandım. Yorulmadan beni takip edeceğinizi söylüyorsunuz. Fakat arkadaşlar yorulmamak ne demek? Elbette yorulacaksınız. Benim sizden istediğim şey yorulmamak değil, yorulduğunuz zaman dahî durmadan yürümek, yorulduğunuz dakikada dinlenmeden beni takip etmektir. Yorgunluk, her insan için tabii bir hâlettir. Fakat insanda yorgunluğu yenebilecek manevi bir kuvvet vardır ki, bu kuvvet yorulanları dinlendirmeden yürütür.

Sizler, yani yeni nesil Türkiye'nin genç evlatları, yorulsanız dahi beni takip edecekseniz. Ben bu akşam buraya yalnız bunu size anlatmak için gelmiş bulunuyorum. Dinlenmemek üzere yürümeye karar verenler, asla ve asla yorulmazlar. Türk gençliği gayeye, bizim yüksek idealimize durmadan, yorulmadan yürüyecektir.”



2019, Murat YÜCEDAĞ

Türkiye



Değerli okurlarım kitamıza başlamadan önce bir ricam olacak.

Kitapta hata gördüğünüz de hemen kızmayın, sizlerin de desteğiyle hatalarımızı ve eksikliklerimizi tarafıma bildirmeniz durumunda bir sonraki baskıda düzeltilecektir. :)

Murat YÜCEDAĞ

muratyucedag.wordpress.com

facebook.com/murattyucedag

twitter.com/murattyucedag

linkedin.com/in/murat-yücedağ-186933149

udemy.com/user/murat-yucedag-3

youtube.com/user/YazilimHerYerde

github.com/MuratYucedag/testproje

yucedagmurat23@gmail.com



Kitaba dair örnek uygulama ve kaynak kodları aşağıdaki QR kod ile indirebilirsiniz...



İÇİNDEKİLER

BÖLÜM 1: GİRİŞ	1
Neden Yazılım?	2
Nedir Bu Algoritma?	3
Yazılım Nedir?	5
Kodlama Nedir?	6
C# Programlama Dili Hakkında	7
Neler Öğrendik?	9
BÖLÜM 2: VISUAL STUDIO	11
Derleyici Nedir?	12
Visual Studio	12
İndirme ve Kurulum	12
Neler Öğrendik?	15
BÖLÜM 3: İLK PROJEMİZ	17
Yeni Proje Oluşturma	18
Neler Öğrendik?	23
BÖLÜM 4: TEMEL ARAÇLARIN KULLANIMLARI	25
Araç Nedir?	26
Button Aracı	26
Label Aracı	34
TextBox Aracı	37
ComboBox Aracı	40
ListBox Aracı	42
PictureBox Aracı	45
Neler Öğrendik?	50

BÖLÜM 5: DEĞİŞKENLER	53
Değişken Nedir?	54
string Değişkenler	55
int Değişkenler	57
double Değişkenler	68
Neler Öğrendik?	75
BÖLÜM 6: KARAR YAPILARI	77
Karar Yapısı Nedir?	78
Timer Aracı	92
Neler Öğrendik?	98
BÖLÜM 7: DÖNGÜLER	101
Döngü Nedir?	102
For Döngüsü	102
While Döngüsü	108
Neler Öğrendik?	118
BÖLÜM 8: DİZİLER	121
Dizi Nedir?	122
Foreach Döngüsü	131
Neler Öğrendik?	135
BÖLÜM 9: RANDOM YAPISI	137
Random Nedir?	138
Neler Öğrendik?	151
BÖLÜM 10: FORMLAR ARASI GEÇİŞ VE PROJE	153
Formlar Arası Geçiş İşlemi	154
Neler Öğrendik?	163

1

GİRİŞ

BU BÖLÜMDE

Neden Yazılım?	2
Nedir Bu Algoritma?	3
Yazılım Nedir?	5
Kodlama Nedir?	6
C# Programlama Dili Hakkında	7
Neler Öğrendik?	9

Kitabımızın ilk bölümünde; Neden Yazılım? başlığıyla başlayarak yazılım, kodlama, algoritma gibi teorik kavramları, C# programlama dili ile neler yapılabileceğini ve algoritmanın nasıl oluşturulacağına dair bilgiler öğreneceğiz.

NEDEN YAZILIM?

Gençlik yıllarımızdan birkaç yıl öncesine kadar anne ve babalarımız bizlere hep sitem ederdi; **O telefonu bırak artık eline yapıştı!** diye. Bu cümleyi kuran anne babalar bugün sosyal medyada bizlerden yani çocuklarından daha fazla zaman geçiriyorlar.



Günümüzde akıllı cihazların girmedığı ev, internetin olmadığı ortam neredeyse yok gibi. Eskiden saatlerce hatta günlerce süren ve tamamlanması için çok ciddi enerji harcanan işlemleri artık dakikalar içinde birkaç tıklama ile yapabiliyoruz. Örneğin; geçtiğimiz yıllarda faturalarımızı yatırmak için onlarca dakika sırada beklerken artık telefon ya da bilgisayar üzerinden yalnızca 1-2 dakika gibi kısa bir sürede faturalarımızı yatırabiliyoruz. Birkaç yıl önce yemek yemek için dışarıda bir lokantaya giderken şimdi internet üzerinden istediğimiz yemeği bir tıkla sipariş edebiliyoruz. İşte tüm bunlar yazılım sayesinde gerçekleşiyor.

Bizim toplumumuzda en kıymetli üniversite bölümü doktorluk yani tıptır. Ancak son yıllarda özellikle **Bilişim & Yazılım** tabanlı mühendislik bölümleri tıp kadar kıymetli olmaya başladı. Bizce tıp ne kadar değerli ise yazılım da o kadar değerlidir. Çünkü tıp cihazlarının çalışması için yazılıma ihtiyaç vardır. Yani tıp ve yazılım artık birbirini tamamlayan iki parça gibi oldular.



2018/2019 yıllarının en değerli markalarının yer aldığı listeler yayınlanmaktadır. Bu listelere Google'dan ulaşabilirsiniz. Dünya üzerindeki tüm sektörler arasında yapılan marka kıyaslamalarda (turizm, ticaret, sanayi, sağlık, gıda, tekstil...) son zamanların en popüler firmaları; Amazon, Apple, Microsoft, Google, Samsung gibi firmalar yüzbinlerce belki de daha fazla şirketin olduğu listede ilk 5 içinde yer almaktadırlar.

Bu 5 firmanın ortak özelliği 5'inin de Yazılım & Bilişim firması olmasıdır. Bu listenin üst sıralarında yazılım firması dışında herhangi bir firma bulunmamaktadır. Listenin ortalarına doğru gıda firmaları, akaryakıt firmaları gibi şirketler yer almaya başlıyor.

Kaynak;

<https://www.brandingturkiye.com/dunyanin-en-degerli-500-markasi-2019-tam-liste/>

Yaptığınız küçük bir mobil uygulamayı Google markete ya da AppStore gibi mağazalara yükleyerek insanlara ulaştırabilir ve onların yaşamlarını kolaylaştırabilirsiniz.

Annelerimizin yemekleri her zaman çok lezzetlidir. Son zamanlarda ise çocuklarına daha lezziz yemekler yapmak için onlar da mobil cihazlardan yararlanarak yemek sitelerine, yemek videolarına göz atarak yeni tarifler deniyorlar ve çocukları için çok güzel yemekler yapıyorlar. Annemiz yemeği hazırladıktan sonra tabakları taşımaya yardım edelim tüm işi ona bırakmayalım çünkü annelerin en sevdiği çocuklar onlara minik yardımlar yapan çocuklardır. Bir önemli not da **1. Bölümü** bitirdikten sonra internette bir yemek tarifi sitesine girip beğendiğimiz bir yemeği annemize gösterelim ve bu akşam beğendiğiniz o yemeği annenizden yapmasını isteyin eğer siz de yardım edeceğinizi sözlerseniz anneniz size o yemeği mutlaka yapacaktır. :)

NEDİR BU ALGORİTMA?

Algoritma kavramını pek çok yerde duymuş olabilirsiniz.

Peki, nedir bu algoritma?

Bir problemin çözümünde kullanılan adımların tamamıdır şeklinde bir tanım yapabiliriz. Aslında sadece bilgisayar dünyası için değil günlük hayatta da her an karşımıza çıkan ve kullanılan bir kavramdır. Verdiğimiz her karar bir algoritma ürünüdür.

2

Visual Studio

BU BÖLÜMDE

Derleyici Nedir? 12

Visual Studio 12

Neler Öğrendik? 15

Kitabımızın bu bölümünde kodlama yaparken kullanacağımız program olan **Visual Studio** programının ne olduğunu ve kurulumunun nasıl yapılacağını öğreneceğiz.

DERLEYİCİ NEDİR?

Programlama dillerine ait kodları yazmamız için gerekli olan ortamlara derleyici denir. Derleyiciler yazmış olduğumuz kodları bilgisayarın anlayabileceği 0 ve 1'den oluşan ikili sayı sistemine dönüştürürler. Bilgisayarlardaki bütün komutlar ikili sayı sistemi prensibine dayanır. İkili sayı sistemleri hakkında daha detaylı bilgi için kısa bir Google araştırması yapabilirsiniz. Her dilin kendine ait bir derleyici ortamı vardır. Kitabımızda kullanacağımız derleyici **Visual Studio** olacaktır.



VISUAL STUDIO

Microsoft tarafından geliştirilen bir geliştirme ortamıdır. Visual Studio aracılığı ile masa üstü uygulamalar, mobil uygulamalar, web siteleri, tasarım ara yüzleri gibi şeyleri geliştirebilirsiniz. Visual Studio içerisinde C#, Visual Basic, C++ gibi daha pek çok programlama dilleri bulunmakta ve geliştirme yapılabilmektedir.

İNDİRME VE KURULUM

Visual Studio'nun 2017 ya da 2019 sürümü ile çalışabiliriz. 2019 sürümünü indirmek için Microsoft'un aşağıda bulunan sitesi üzerinden Community sürümünü indirebiliriz.

← ↻ 🌐 visualstudio.microsoft.com İndirmeler | IDE, Code ve Team Foundation Server | Visual Studio

Microsoft | Visual Studio [Görünler](#) [Karşından Yüklemler](#) [Market](#) [Destek](#) [Abone Erişimi](#) [Ücretsiz Visual Studio](#) Tüm Microsoft [🔍](#) [👤](#)

Karşıdan Yüklemler

Windows macOS

Visual Studio 2019

Android, iOS, Windows, web ve bulut için tam özellikli tümleşik geliştirme ortamı (IDE)

Community
Güçlü IDE, öğrenciler için açık kaynak katılda bulunanlar ve bireysel kullanıcılar için uygundur.

Professional
Profesyonel IDE, küçük ekipler için en uygun seçenek.

Enterprise
Herhangi boyutta olursa olsun tüm ekipler için ölçeklenebilir, uçtan uca çözüm.

Sürüm notları

Ücretsiz indir

Sürümleri karşılaştır
Çevrimdışı yükleme

Visual Studio Code

İhtiyaçlarınızı uyum sağlayan hızlı, ücretsiz ve açık kaynak kodlu düzenleyicisi

Sürüm notları

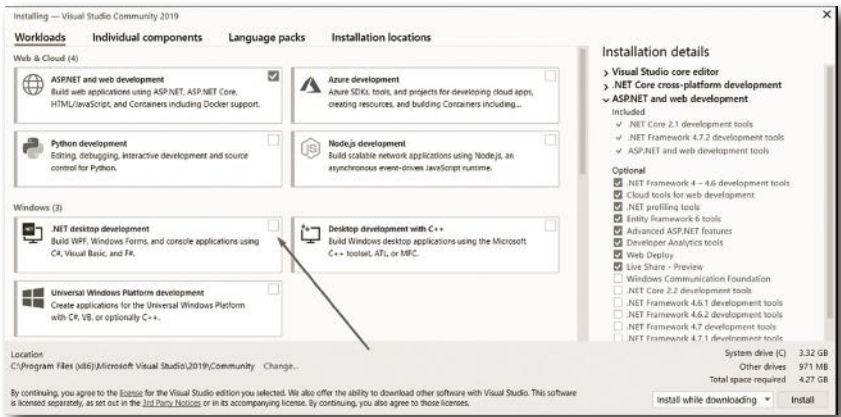
Ücretsiz indir

Visual Studio Code'u indirip kullandığınızda lisans koşullarını ve gizlilik bilginizi

İndirme işlemi tamamlandıca indirdiğimiz dosyayı açıp kurulumu başlatalım.



Karşımıza çıkan dil seçeneklerinden okla gösterilen .net masa üstü geliştirme seçeneğine seçelim ve hemen üzerinde bulunan 2. sekmeye tıklayalım.



Açılan pencerede seçili alanları sabit bırakıp herhangi bir değişiklik yapmadan 3. sekmeye geçelim.



3

İLK PROJEMİZ

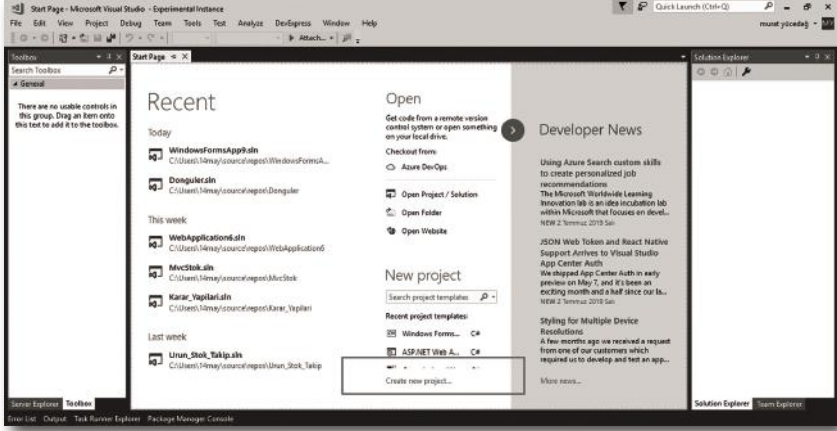
BU BÖLÜMDE

Yeni Proje Oluřturma	18
Neler Öğrendik?	23

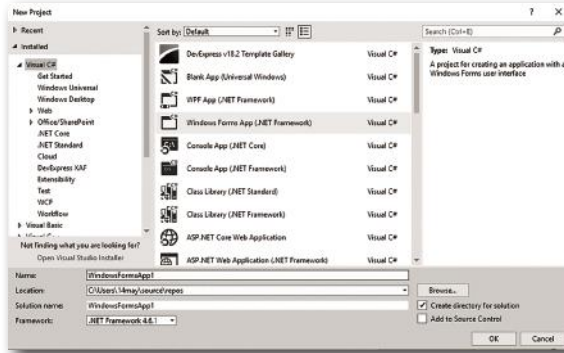
Kitabımızın bu bölümünde Visual Studio üzerinde yeni bir projenin nasıl oluşturulacağı, projelerin nasıl çalıştırılıp nasıl durdurulacağı, projemizin form, araç kutusu, özellikler penceresi ve çözüm penceresi gibi alanlarının ne işe yaradığını, formumuzun özelliklerinin nasıl değiştirileceğini öğrenip ilk projemizi oluşturacağız.

YENİ PROJE OLUŞTURMA

Visual Studio'da yeni bir proje oluşturma işlemi yapalım. Bunun için sağ alt tarafta bulunan **Create New Project** yani (Yeni Proje Oluştur) seçeneğine tıklayarak başlayalım.

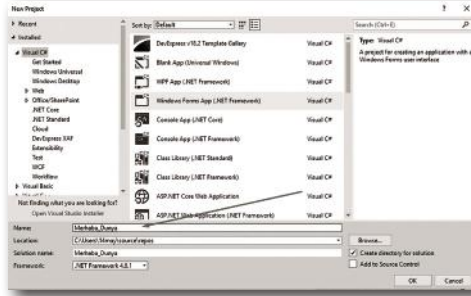


Açılan pencerenin sol tarafında **Visual Studio** üzerinden geliştirme yapabileceğimiz platform ve programlama dilleri mevcuttur. Buradan **C#** dilini seçelim. Visual **C#** seçeneğini seçtikten sonra **C#**'in de kendi içerisinde bazı başlıkları bulunmaktadır. Bunlar arasında bizim proje geliştireceğimiz kısım **Form** alanıdır. Form kısmı aracılığı ile ara yüz tasarımı yapabilir araç kullanabiliriz. Ara yüz tasarımı ve araç kullanımlarını bir sonraki bölümde göreceğiz.

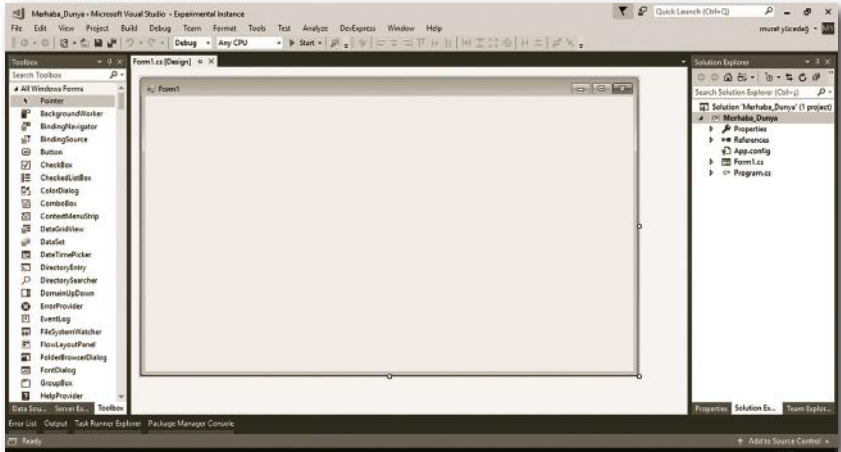


Daha sonra alt kısımda bulunan **Name** alanından projemizin ismini değiştirelim. Projelerimize isim verirken elimizden geldiğince **Türkçe** karakter kullanmaktan kaçınalım. Türkçe karakter kullanımı Visual Studio içerisinde herhangi

bir hata oluşturmayacaktır ancak programlama dillerinin evrensel dili İngilizce olduğu için Türkçe karakter barındırmayan isimler verelim. Ayrıca projelerimizi isimlendirirken eğer 2 ya da daha fazla kelimedenden oluşuyorsa aralarına boşluk koymak yerine **alt tre _** gibi bir sembolle birleştirerek yazarsak biraz daha profesyonel bir kullanım olacaktır. Dünya genelinde ilk ders cümlecığı **Hello World** yani **Merhaba Dünya** olarak kabul edilmiştir. Bundan dolayı bizde ilk projemizin ismini **Merhaba_Dunya** koyabiliriz. Daha sonra **Ok** butonuna tıklayalım.



Karşımıza aşağıdaki gibi bir pencere gelecektir. Bu pencereyi 5 temel kategoriye ayırabiliriz. Şimdi bu 5 başlığı inceleyelim.



- » İlk olarak üst kısımda bulunan Menü alanı: buradan yeni projeler oluşturabilir, araç kutumuzu, özellikler pencremizi ve diğer diyalog pencerelerimizi açabilir, projemizi derleyebilir, Visual Studio ayarlarını değiştirebiliriz.



4

TEMEL ARAÇLARIN KULLANIMLARI

BU BÖLÜMDE

Araç Nedir?	26
Neler Öğrendik?	50

Bu bölümde araç kavramının ne olduğu, araçların özelliklerinin nasıl değiştirileceği, araçlarımızın form üzerine eklemelerinin nasıl yapılacağı, en sık kullanacağımız araçlar olan Button, TextBox, ComboBox, ListBox, PictureBox, Label, Radiobutton ve CheckBox araçlarının nasıl kullanılacağını öğrenip bu araçlarımızla projeler geliştireceğiz.

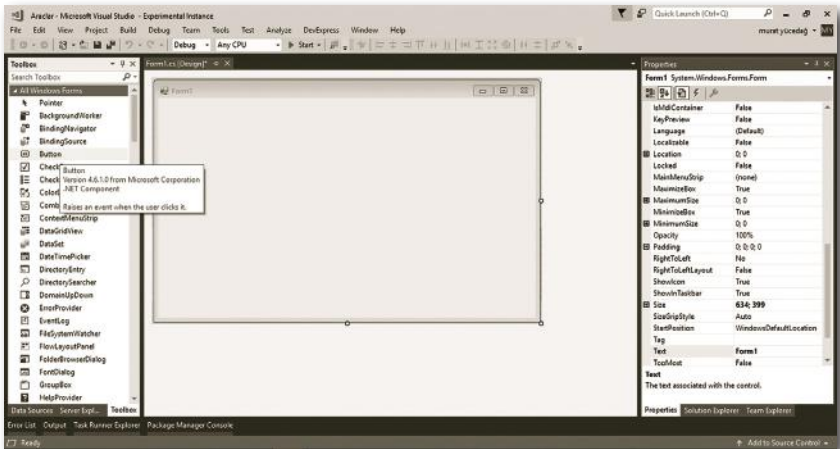
ARAÇ NEDİR?

Bilgisayar kullanmayı bilen herkes mutlaka bir şekilde **paint** programı ile çizim yapmıştır. Paint programında çizim yapacağımız boş beyaz alanın üst kısmında birtakım araçlar mevcuttur. Bunlar çizimi kolaylaştırmak ya da istediğimiz bir geometrik şekli (dörtgen, daire, yıldız, kare vs.) kolaylıkla oluşturabilmek için bize sunulan araçlardır. Bu araçlar sayesinde istediğimiz şekilleri daha düzgün çizebiliriz.

Bunun dışında ayrıca renk seçimi, kalem türü, fırça boyutu gibi çizim yaparken bizlere yardımcı olacak pek çok araç bulunmaktadır. Bu araçlar genel olarak birbirinden farklı işlemleri yapmaktadır. Bir araya geldiklerinde ise çizimin tamamını oluşturan anlamlı bir bütün olurlar. Şimdi C# form projelerimizde en çok kullanacağımız bazı araçlarımızı tek tek ele alalım.

BUTTON ARACI

Button kelimesi Türkçeye **Düğme** olarak çevireceğimiz ve görev olarak da tıpkı bir düğmenin yaptığı işleri yapan aracımızdır. Evlerimizde oturduğumuz odalarımızın lambalarını açmak için nasıl ki bir düğmeye ihtiyaç duyuyorsak; aynı şekilde programlarımızda da bir işlemi gerçekleştirmek için düğmeye ihtiyacımız vardır. Şimdi formumuza bir adet Button aracı ekleyip uygulama üzerinde kullanarak daha somut bir şekilde ne işe yaradığını görelim. Açmış olduğumuz boş formumuza sol tarafta bulunan **Araç** kutumuzdan bir adet Button aracını sürükleyerek formumuza bırakıyoruz.



Butonumuzu formumuzun içerisine ekledikten sonra özellikler penceresini açarak bu aracımızın üzerindeki yazısı, yazı boyutu, arka plan rengi gibi bazı özelliklerini değiştirelim. Özellikler penceresini açmak için aracımıza bir kere tıklayıp daha sonra **F4** tuşuna basabiliriz. Ya da aracımızın üstüne sağ tuşla tıklayarak en altta bulunan **Özellikler** yani İngilizce karşılığı ile **Properties** seçeneğine tıklayabiliriz.



Özellikler penceresini açtıktan sonra yanda bulunan görselde kutucuk içerisine aldığımız A-Z alanına tıklayarak özellikleri alfabetik sıraya göre dizelim.



Butonumuzun bazı özelliklerini görelim.

- » **Back Color:** Butonumuzun arka plan renginin değiştirileceği alandır.
- » **Back Gorund Image:** Butonumuzun arka planına resim eklemek için kullanılacak alanımızdır.
- » **Cursor:** Butonumuzun üzerine geldiğimiz zaman fare imlecinin nasıl olması gerektiğinin belirlendiği özelliktir.
- » **Enabled:** Aktiflik-pasiflik özelliğidir. Butonumuzun enabled özelliği false olursa butonumuzu tıklanamaz hale gelir.
- » **Font:** Butonumuzun yazı biçiminin düzenleneceği alandır.
- » **Location:** Butonumuzun form üzerinde bulunan konumunun yer aldığı alandır.
- » **Size:** Butonumuzun boyutunun belirlendiği alandır.
- » **Text:** Butonumuzun üzerinde yazan yazının yer aldığı alandır.
- » **Visible:** Butonumuzu görünmez hale getirmek için kullanılacak alandır.

8

DİZİLER

BU BÖLÜMDE

Dizi Nedir?	122
Neler Öğrendik?	135

Bu bölümde dizi kavramını, neden dizi kullandığımızı, nasıl dizi tanımlandığını, foreach döngüsünü, kodlarla PictureBox aracımıza nasıl resim ekleneceğini, diziler içerisinde aritmetik işlem yapmayı öğreneceğiz.

Dizi NEDİR?

Geldik kitabımızın son konusuna. Birbiriyle aynı türde olan veri topluluklarına **dizi** denir. Renkler, sayılar, bir sınıfta bulunan öğrenciler, hayvanlar, şehirler, kuşlar hepsi birer dizidir. Diziler bilgisayarımızın geçici belleği olan ram bellekte tutulurlar. Bir yapının dizi olduğunu anlamak için [] sembolü içermesi gerekmektedir. Birden fazla dizi oluşturma yöntemi vardır.

Kitabımız yeni başlayanlar ve çocuklar için hazırlandığı için biz burada en basit yöntemleri tercih edeceğiz. Dizilerde sayma işlemleri 0'dan başlar. Neden dizi kullanırız diye soracak olursak bazı durumlarda değişkenler yeterli olmayacaktır.

Örneğin; futbol müsabakalarında yapılan kura çekimleri için takımların isminin bir torbaya konulup kura çekimi yapılması bir dizi yöntemiyle kolaylıkla sağlanabilir veya sınıftaki öğrencilerin temizlik sırası için öğrenci isim ve soy isimleri bir dizi içerisine eklenip bu dizi sırası takip edilerek temizlik işlemi sırayla yapılabilir. Dizilerin söz dizimine bakacak olursak;

Değişken Türü [] Dizi Adı = { Elemanlar }

şeklinde olacaktır.

Dizilerle ilgili isim verirken değişken adlandırmada geçerli olan kurallar burada da geçerlidir. Örneğin; dizi adı ayrı olamaz, sayı ile başlayamaz gibi... Dizi adlarını da tıpkı değişken isimlerinde oldukça doğru ve yalın isimlerden seçmeliyiz. Birkaç tane dizi örneği yazalım.

Kuşlar dizisi: serçe, kanarya, bıldırcın, şahin, karga, papağan...

Sayılar dizisi: 4, 85, 36, 1881, 19, 24, 125, 852...

Renkler dizisi: mavi, yeşil, beyaz, turuncu, pembe, kahverengi, yeşil...

Süper kahramanlar dizisi: Superman, Batman, Spiderman, Hulk, Captain Marvel...

Şehirler dizisi: Ankara, İstanbul, Bursa, Adana, Elazığ...

Bu şekilde örneklerimiz arttırılabilir. Şimdi uygulamalarımızı kodlar üzerinden gerçekleştirelim.

Uygulama 1

Arkadaşlarımızın isimlerini tutan bir dizi oluşturup bu dizinin 3 numaralı index'te bulunan elemanını yazdıralım.

```
private void button1_Click(object sender, EventArgs e)
{
    string[] arkadaslar = { "Mehmet", "Bahadır", "Mesut", "Didem", "Funda" };
    label1.Text = arkadaslar[3];
}
```



arkadaslar isminde string türünde bir dizi oluşturduk. Bu dizimize süslü parantez blokları içerisinde 5 tane değer girişi yaptık. Bu değerlerin bellekte sıralanışı şöyle olacaktır;

0	Mehmet
1	Bahadır
2	Mesut
3	Didem
4	Funda

Dizilerde sayma & sıralama işlemleri 0'dan başladığı için Mehmet 0 nolu index'te yani sırada, Funda ise 4 nolu index'te yer alacaktır. Label aracımıza arkadaşlar dizimizin 3 nolu index'inde bulunan değeri yazdırmak istedik. 3 nolu indeximizde Didem bulunduğu için bu değeri Label aracımıza yazdırarak kod bloğumuzu tamamladık.

Uygulama 2

Ülkelerin isimlerini tutan 7 elemanlı bir dizi oluşturalım.

```
private void button1_Click(object sender, EventArgs e)
{
    string[] ulkeler = { "Türkiye", "Fransa", "Almanya", "Belçika",
    "Hollanda", "İtalya", "Polonya" };
}
```

10

FORMLAR ARASI GEÇİŞ VE PROJE

BU BÖLÜMDE

Formlar Arası Geçiş İşlemi
Neler Öğrendik?

154
163

Bu bölümde projemize nasıl yeni form eklendiğini, formlar arası geçişin nasıl yapıldığını, projenin nasıl farklı bir formdan çalıştırıldığını ve kullanıcı adı & şifre kontrolü yapılarak eğer bilgiler doğruysa yapmış olduğumuz projenin ana ekranına nasıl ulaşacağımızı öğreneceğiz.

FORMLAR ARASI GEÇİŞ İŞLEMİ

C# formda bazen projemizde birden fazla pencere olması gerekebilir. Örneğin bir pencerede öğrencilere ait sınav notları gözükrken bir başka pencerede öğrencinin haftalık ders programı gözükebilir. İşte böyle durumlarda projemize yeni formlar ekleyip genişletebiliriz. Yeni bir tane proje oluşturalım ve bu projemizde bir formdan diğerine Button yardımıyla nasıl geçiş yaptığımızı görelim.

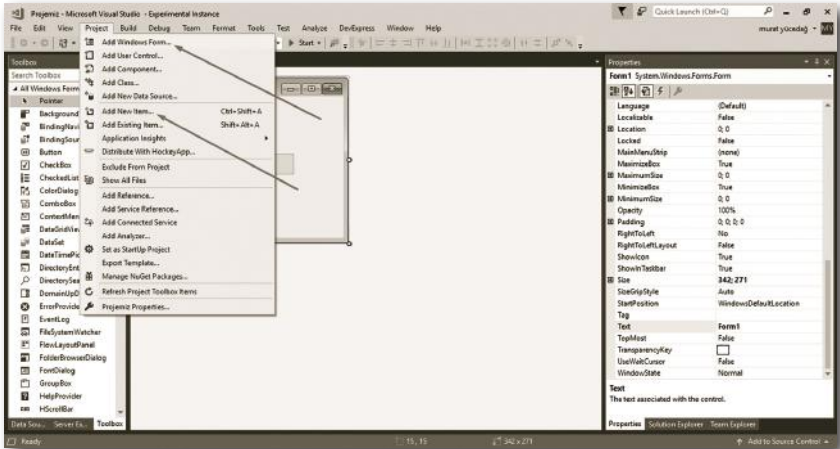
Uygulama 1

1. formdan 2. forma geçiş yapılan projemizi hazırlayalım.

İlk olarak yeni bir tane proje oluşturup formumuzun içine Button aracık ekleyerek başlayalım. Butonumuzun ismi "Form 2'ye Git" olsun.



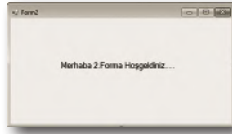
Şimdi projemize yeni bir form ekleyeceğiz. Bunun için yukarıda bulunan **Project** menüsüne tıklalayalım. Açılan pencerede **Add Windows Form...** yani Windows formu ekle ya da **Add New Item...** yani yeni öge ekle seçeneklerinden herhangi birine tıklayarak form ekleme penceremizi açabiliriz. Her iki alandan da aynı noktaya ulaşacağız.



Açılan pencereden **Windows Form** seçeneğini seçip **Add** yani ekle kısmına tıklayalım. Formumuzun ismi başlangıçta **form2** olarak geldi isterseniz bunu değiştirebilirsiniz.



Oluşturduğumuz yeni formumuza bir tane Label aracı ekleyelim ve bu Label aracımıza istediğimiz herhangi bir metni **Özellikler** penceresinden değiştirerek yazalım.



Tasarımımızı tamamladığımıza göre tekrar 1. forma gidip **Button** aracımıza tıklayıp kodlarımızı yazalım.

Örnek kod

```
private void button1_Click(object sender, EventArgs e)
{
    Form2 frm = new Form2();
    frm.Show();
}
```



MURAT YÜCEDAĞ

muratyucedag.wordpress.com



facebook.com/murattyucedag



twitter.com/murattyucedag



linkedin.com/in/murat-yücedağ-186933149



udemy.com/user/murat-yucedag-3



youtube.com/user/YazilimHerYerde



github.com/MuratYucedag/testproje



yucedagmurat23@gmail.com

