

Uygulama Geliştirerek C#

11 Ekim 2020 Pazar

16:42

- <https://www.udemy.com/course/sifirdan-ileri-seviye-csharp-programlama/>

- ☒ [01 Temel Araç Kullanımları](#)
- ☒ [02 Değişkenler](#)
- ☒ [03 Karar Yapıları](#)
- ☒ [04 Döngüler](#)
- ☒ [05 Diziler](#)
- ☒ [06 Metotlar](#)
- ☒ [07 Sınıflar](#)
- ☒ [08 Ekstra Araçlar](#)
- ☒ [09 Ekstra Yapılar](#)
- ☒ [10 Dosya İşlemleri](#)
- ☒ [11 Hata Kontrolleri](#)



CENGİZHAN TOPÇU

MEKATRONİK MÜHENDİSİ

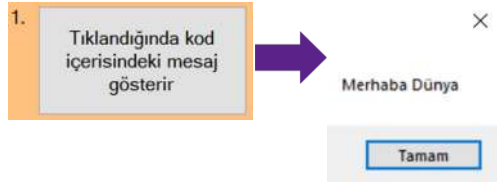
01 Temel Araç Kullanımları

3 Ekim 2020 Cumartesi 19:32

01 Temel Araç Kullanımları

3. Merhaba Dünya

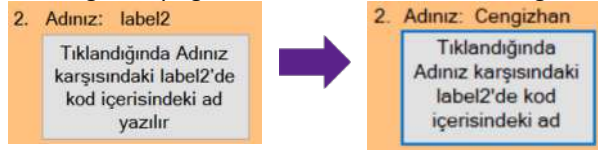
- Kod yazarken önümüze yanına gelebilecek komutlar çıkar. Bu çıkan sonuçların yanlarında küp sembolü verirse bu komuttan sonra parantez gelir.



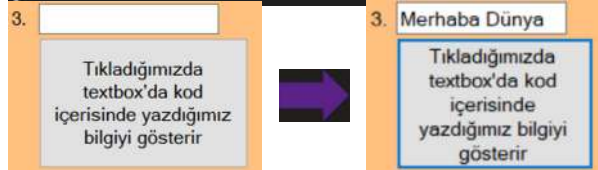
```
private void button1_Click(object sender, EventArgs e)
{
    MessageBox.Show("Merhaba Dünya");
}
```

4. Label ve Textbox Kullanımı

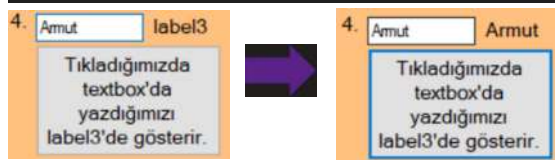
- Bilgi nereye gösterilecekse o sol tarafa, o bilgi nereden yazılacaksa o da sağ tarafa yazılır.



```
private void button2_Click(object sender, EventArgs e)
{
    label2.Text = "Cengizhan";
}
```



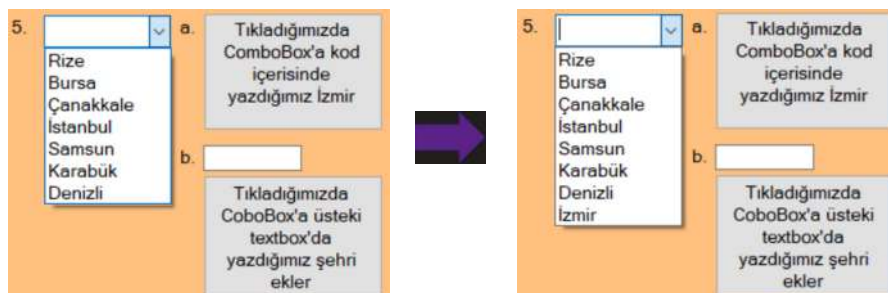
```
private void button3_Click(object sender, EventArgs e)
{
    textBox1.Text = "Merhaba Dünya";
}
```



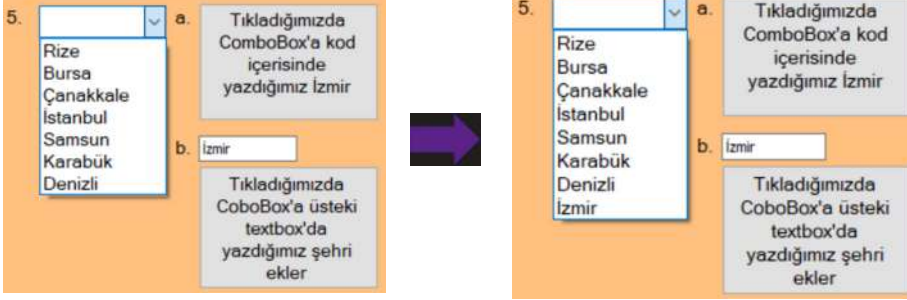
```
private void button4_Click(object sender, EventArgs e)
{
    label3.Text = textBox2.Text;
}
```

5. Combobox ve Listbox Araçları Kullanımı

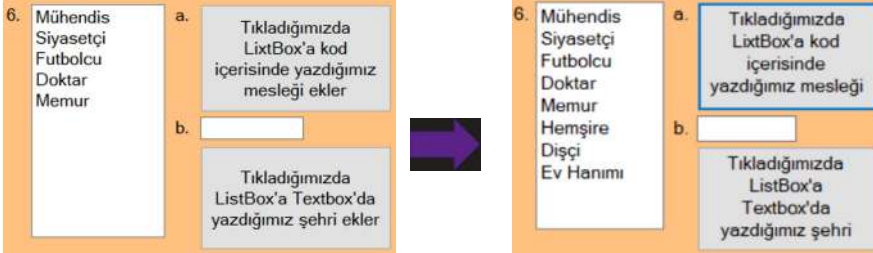
- Listbox'ın Combobox'dan farkı listedekilerin hepsi gözükür. Combobox'a tıkladığımızda liste gözükür.
- Kod satırındaki tırnak işaretini koymamızın nedeni kutuya yazılan bilgidен sonra gelen bilgi arasında boşluk olması için kullanılır.



```
private void button5_Click(object sender, EventArgs e)
{
    comboBox1.Items.Add("İzmir");
}
```



```
private void button6_Click(object sender, EventArgs e)
{
    comboBox1.Items.Add(textBox3.Text);
}
```



```
private void button7_Click(object sender, EventArgs e)
{
    listBox1.Items.Add("Hemşire");
    listBox1.Items.Add("Dişçi");
    listBox1.Items.Add("Ev Hanımı");
}
```



6. Maskedtextbox - PictureBox - Datetimepicker Kullanımları

- Kod satırındaki tırnak işaretlerini koymamızın nedeni yazılan bilgidan sonra gelen bilgi arasında boşluk olması için kullanılır.



```
private void button8_Click(object sender, EventArgs e)
{
    listBox2.Items.Add(dateTimePicker1.Text + " " + maskedTextBox1.Text + " " + maskedTextBox2.Text + " " + maskedTextBox3.Text);
}
```

7. Checkbox - Radiobutton - Richtextbox - Panel ve Groupbox Kullanımları

- Checkbox kutuların hepsini seçebiliyorken radiobutton'da sadece biri seçilebilir.
- Panel ve GroupBox içerisine araç eklenebilir. Panel'in GrouPbox'dan tek farkı kenar çizgileri yoktur, rengini değiştirdiğimizde net olarak gözükür.
- Richtextbox'ın textbox'dan farkı biraz daha genişletilmiş hali yani alt satıra inilebiliyor.

8. Örnek Proje - Login Paneli Oluşturma

- Yazılan bilginin şifreli görünümünde olması için PasswordChar'ı True yaparız.

9. Örnek Proje - Uçak Bilet Rezervasyon Sistemi

- listbox'a kaydırma çubuğu eklenmesi için HorizontalScrollbar True yapılır.
- Label'ı gizlemek için görünürlüğü kapatmak için Visible'u false yaparız.

```
private void button10_Click(object sender, EventArgs e)
{
    Form2 frm = new Form2();
    frm.Show();
}
```

```
private void button1_Click(object sender, EventArgs e)
{

```

```
    listBox1.Items.Add("Rotanız:" + comboBox1.Text + "-" + comboBox2.Text + " Tarih:" + dateTimePicker1.Text + " Saat:" +
maskedTextBox1.Text + " Yolcu Bilgileri - Adı:" + textBox1.Text + " TC:" + maskedTextBox2.Text + " Telefon:" + maskedTextBox3.Text);
    MessageBox.Show("Başarıyla Oluşturulmuştur. Sizlere İyi Yolculuklar Dileriz");
}

```

```
private void button2_Click(object sender, EventArgs e)
{

```

```
    label10.Text = comboBox2.Text;
    comboBox2.Text = comboBox1.Text;
    comboBox1.Text = label10.Text;
}

```

02 Değişkenler

3 Ekim 2020 Cumartesi 19:33

02 Değişkenler

10. Değişken Nedir & Teorik Kavramlar

11. String Değişkenler

- Değişken atarken önce yazdığımıza uygun değişken türünü yazar sonra değişken adını yazarız.

1. label1

Tıkladığımızda değişken atadığımıza yazılanı label'da yazar

1. Fenerbahçe

Tıkladığımızda değişken atadığımıza yazılanı label'da yazar

```
private void button1_Click(object sender, EventArgs e)
{
    string metin;
    metin = "Fenerbahçe";
    label1.Text = metin;
}
```

2. Adınız: Soyadınız:

Tıkladığımızda değişken atadığımıza yazılanı textbox'a yazar

2. Adınız: Cengizhan Soyadınız: Topçu

Tıkladığımızda değişken atadığımıza yazılanı textbox'a yazar

```
private void button2_Click(object sender, EventArgs e)
{
    string Ad,Soyad;
    Ad = "Cengizhan";
    Soyad = "Topçu";
    textBox1.Text = Ad;
    textBox2.Text = Soyad;
}
```

3. Cengizhan label6

Tıkladığımızda textbox üzerine yazdığımızı label'a yazar

3. Cengizhan Cengizhan

Tıkladığımızda textbox üzerine yazdığımızı label'a yazar

```
private void button3_Click(object sender, EventArgs e)
{
    string isim;
    isim = textBox3.Text;
    label6.Text = isim;
}
```

12. Int Değişkenler

- C#'da araçların çoğunluğu string formattadır. Liste araçları string formatta değildir. listBox, comboBox gibi araçlar istenilen değişken formatında kullanılabilir.
- Int değişkenini tam sayılarda kullanıldığından label, textbox, maskedTextBox gibi araçlara sayıyı atayabilmemiz için stringe çevirmemiz gerekir. Bunun için ToString kullanılır.
- String formattaki araca ifade yazdırırken int değişkeni yanında string ifade var ise ToString dönüşümü uygulanmaz.

4. label9

Tıkladığımızda label üzerinde sayı yazar

4. 12

Tıkladığımızda label üzerinde sayı yazar

```
private void button4_Click(object sender, EventArgs e)
{
    int sayi;
    sayi = 12;
    label9.Text = sayi.ToString();
}
```

5. label11

Tıkladığımızda labelde iki değişkenin toplamını verir

5. 10

Tıkladığımızda labelde iki değişkenin toplamını verir


```
private void button5_Click(object sender, EventArgs e)
{
    int sayi1, sayi2, toplam;
    sayi1 = 5;
    sayi2 = 5;
    toplam = sayi1 + sayi2;
    label11.Text = toplam.ToString();
}
```

6. label13

Tıkladığımızda karenin bir kenarını kodda yazan ve label'de alanı ve çevreyi verir



6. Alan:20 Çevre:40

Tıkladığımızda karenin bir kenarını kodda yazan ve label'de alanı ve çevreyi verir

```
private void button6_Click(object sender, EventArgs e)
{
    int kenar, alan, cevre;
    kenar = 10;
    alan = kenar * 2;
    cevre = kenar * 4;
    label13.Text = "Alan:" + alan + " Çevre:" + cevre;
}
```

7. label15

Tıkladığımızda bir öğrencinin üç sınav notun ortalamasını hesaplayıp label'da yazan program



7. Ders Ort: 61

Tıkladığımızda bir öğrencinin üç sınav notun ortalamasını hesaplayıp label'da yazan program

```
private void button7_Click(object sender, EventArgs e)
{
    int not1, not2, not3, ortalama;
    not1 = 55;
    not2 = 70;
    not3 = 60;
    ortalama = (not1 + not2 + not3) / 3;
    label15.Text = "Ders Ort: " + ortalama;
}
```

13. Int Değişkenler / Klavyeden Veri Girişi

- Textbox, string aracı olduğu için klavyeden sayı girebilmemiz için dönüştürme işlemi yapmamız gerekir. Bunu için convert kullanılır. Convert kullandıktan sonra yanına dönüştürecekimiz aracı gireriz. Bunun için kullanabileceğimiz araçlar.ToInt16, ToInt32, ToInt64'dür.

8. 5

label17

Tıkladığımızda textbox'a yazılan tam sayı değerini label'a yazar



8. 5

125

Tıkladığımızda textbox'a yazılan tam sayı değerini label'a yazar

```
private void button8_Click(object sender, EventArgs e)
{
    //int x;
    //x = Convert.ToInt32(textBox4.Text);
    //label17.Text = x.ToString();
    int a, b;
    a = Convert.ToInt32(textBox4.Text);
    b = a * a * a;
    label17.Text = b.ToString();
}
```

14. Int Değişkenler Pratik Örnek Aritmetik 4 İşlem

- Mesaj kutusunda yazdıklarımızı alt satıra geçirmek için "\n" kullanırız.

9. Birinci Sayı: 5

İkinci Sayı: 4

Tıkladığımızda textbox'da yazdığımız sayı değerlerini mesaj kutusunda dört işlem olarak gösterir



Toplam:9
Fark:1
Çarpım:20
Bölüm:1

Tamam

```
private void button9_Click(object sender, EventArgs e)
{
    int a, b, toplama, cikarma, carpma, bolme;
    a = Convert.ToInt32(textBox5.Text);
    b = Convert.ToInt32(textBox6.Text);
    toplama = a + b;
    cikarma = a - b;
    carpma = a * b;
    bolme = a / b;
    MessageBox.Show("Toplam:" + toplama + "\n" + "Fark:" + cikarma + "\n" + "Çarpım:" + carpma + "\n" + "Bölüm:" + bolme);
}
```

15. Double Değişkenler

- Sayı değerin basamak sayısını belirlemek için parentez içinde gösterimin nasıl bir şekilde olacağını parantez içinde 0 rakamını kullanarak gireriz.
- Ondaklıkl sayı kod kısmında girerken nokta ile ayırırken form kısmında virgöl ile ayırıyoruz.

10. label22

Tıkladığımızda bir öğrencinin üç sınav notun ortalamasını hesaplayıp ondalıklı şekilde label'da yazan program

10. 69,67

Tıkladığımızda bir öğrencinin üç sınav notun ortalamasını hesaplayıp ondalıklı şekilde label'da yazan program

```
private void button10_Click(object sender, EventArgs e)
{
    double s1, s2, s3, ort;
    s1 = 70;
    s2 = 60;
    s3 = 79;
    ort = (s1 + s2 + s3) / 3;
    //label22.Text = ort.ToString();
    label22.Text = ort.ToString("0.00");
}
```

11. 65,6542

label23

Tıkladığımızda textbox'a yazılan ondalıklı sayı değerini label'a yazar

11. 65,6542

Tıkladığımızda textbox'a yazılan ondalıklı sayı değerini label'a yazar

```
private void button11_Click(object sender, EventArgs e)
{
    double z;
    z = Convert.ToDouble(textBox7.Text);
    label23.Text = z.ToString();
}
```

16. Pratik Örnek - Öğrenci Sınav Notları Hesaplama Uygulaması

12. Adınız: Cengizhan

Soyadınız: Topçu

Vize: 70 Final: 55 Proje: 85

Hesapla

Sevgili öğrencimiz Cengizhan Topçu 70

```
private void button12_Click(object sender, EventArgs e)
{
    string ad, soyad;
    int vize, final, proje;
    double gano;
    ad = textBox8.Text;
    soyad = textBox9.Text;
    vize = Convert.ToInt32(textBox10.Text);
    final = Convert.ToInt32(textBox11.Text);
    proje = Convert.ToInt32(textBox12.Text);
    gano = (vize + final + proje) / 3;
    listBox1.Items.Add("Sevgili öğrencimiz " + ad + " " + soyad + " " + gano);
}
```

17. Char Değişkenler

- Char'da atadığımız değişkeni tek tırnak içerisinde yazarız. Değişken türü farklı olduğu için string değişkenine dönüştürmek için ToString kullanılır.
- Char değişkeni tek karakter uzunluğundadır. F5'e bastıktan sonra textbox içi boşsa yani null ise hata verir. Aynı şekilde birden fazla karakter girersek yine hata verir.

13. label32

Tıkladığımızda değişken atadığımız yazdığımız label'da yazar

13. g

Tıkladığımızda değişken atadığımız yazdığımız label'da yazar

```
private void button13_Click(object sender, EventArgs e)
{
    char takim;
    takim = 'g';
    label32.Text = takim.ToString();
}
```

14. A

label34

Tıkladığımızda textbox'a yazılanı label'a yazar

14. A A

Tıkladığımızda textbox'a yazılanı label'a yazar


```
private void button14_Click(object sender, EventArgs e)
{
    char ders;
    ders = Convert.ToChar(textBox13.Text);
    label34.Text = ders.ToString();
}
```

18. Proje - Sinema Büfe Satış Uygulaması

- Global dışında yazılanlar yani aracın kodunda yazarsak en başa gidip döngü içine girer. Global'de yazılan döngü içine girmez, sabit olarak kalır.
- İmlecimizi bir yere odaklamak için Focus kullanıyoruz.

15.

**Proje - Sinema
Büfe Satış
Uygulaması**



```
private void button15_Click(object sender, EventArgs e)
{
    Form2 frm = new Form2();
    frm.Show();
}
```



```
int kasatutar = 0;
private void button1_Click(object sender, EventArgs e)
{
    int cay, misir, su, bilet, toplam;
    cay = Convert.ToInt32(textBox1.Text);
    misir = Convert.ToInt32(textBox2.Text);
    su = Convert.ToInt32(textBox3.Text);
    bilet = Convert.ToInt32(textBox4.Text);
    toplam = cay * 2 + misir * 5 + su * 1 + bilet * 16;
    label11.Text = toplam.ToString() + " TL";
    kasatutar = kasatutar + toplam;
    label10.Text = kasatutar.ToString() + " TL";
}
private void button2_Click(object sender, EventArgs e)
{
    textBox1.Text = "";
    textBox2.Text = "";
    textBox3.Text = "";
    textBox4.Text = "";
    textBox1.Focus();
}
```

03 Karar Yapıları

4 Ekim 2020 Pazar 16:07

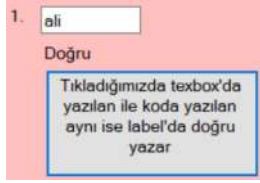
03 Karar Yapıları

19. Operatörler

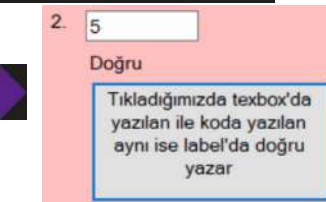
- == Atama, +-* Aritmetik İşlem, == Sorgulama(eşit mi), >= Büyük eşit, <= küçük eşit, != eşit değilse, && Ve, || veya, % Mod

20. If - Else

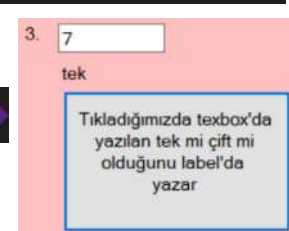
- if yazdıktan sonra tab tuşuna iki kere basarak kendi şablonunu oluşturur.



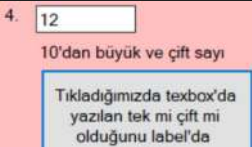
```
private void button1_Click(object sender, EventArgs e)
{
    if (textBox1.Text == "ali")
    {
        label3.Text = "Doğru";
    }
    else
    {
        label3.Text = "Yanlış";
    }
}
```



```
private void button2_Click(object sender, EventArgs e)
{
    int sayi = Convert.ToInt32(textBox2.Text);
    if (sayi==5)
    {
        label4.Text = "Doğru";
    }
    else
    {
        label4.Text = "Yanlış";
    }
}
```



```
private void button3_Click(object sender, EventArgs e)
{
    int a= Convert.ToInt32(textBox3.Text);
    if (a%2==0)
    {
        label1.Text = "çift";
    }
    else
    {
        label1.Text = "tek";
    }
}
```



label7

Tıkladığımızda textbox'da yazılan tek mi çift mi olduğunu label'da yazar

➡

Tıkladığımızda textbox'da yazılan tek mi çift mi olduğunu label'da yazar

```

private void button4_Click(object sender, EventArgs e)
{
    int b = Convert.ToInt32(textBox4.Text);
    if (b%2==0 && b>=10)
    {
        label7.Text = "10'dan büyük ve çift sayı";
    }
    else
    {
        label7.Text = "10'dan büyük değil ya da ve çift değil";
    }
}

```

21. Pratik Örnek - Öğrenci Sınav Durumu Hesaplama

5. Vize: label13

Final: label14

Proje:

Tıkladığımızda textbox'da yazılan sınavların ortalamasına göre dersten geçip geçmediği label'da yazar

➡

5. Vize: 66,83

Final: Geçti

Proje:

Tıkladığımızda textbox'da yazılan sınavların ortalamasına göre dersten geçip geçmediği label'da yazar

```

private void button5_Click(object sender, EventArgs e)
{
    //int vize, final, proje, ortalama;
    //vize = Convert.ToInt32(textBox5.Text);
    //final = Convert.ToInt32(textBox6.Text);
    //proje = Convert.ToInt32(textBox7.Text);
    //ortalama = (vize + final + proje) / 3;
    //label13.Text = ortalama.ToString();
    //if (ortalama >= 50)
    //{
    //    label14.Text = "Geçti";
    //}
    //else
    //{
    //    label14.Text = "Kaldı";
    //}
    double vize, final, proje, ortalama;
    string durum;
    vize = Convert.ToDouble(textBox5.Text);
    final = Convert.ToDouble(textBox6.Text);
    proje = Convert.ToDouble(textBox7.Text);
    ortalama = (vize + final + proje) / 3;
    label13.Text = ortalama.ToString("0.00");
    if (ortalama>=50)
    {
        durum = "Geçti";
    }
    else
    {
        durum = "Kaldı";
    }
    label14.Text = durum;
}

```

22. Pratik Örnek - Alışveriş İndirim Tutar Hesaplama

6. Kitap Adedi:

Tutar: 00 TL

Tıkladığımızda kitapçının textbox'da yazılan kitap adedine göre indirim yapıp fiyatı label'da yazar

➡

6. Kitap Adedi:

Tutar: 96 TL

Tıkladığımızda kitapçının textbox'da yazılan kitap adedine göre indirim yapıp fiyatı label'da yazar

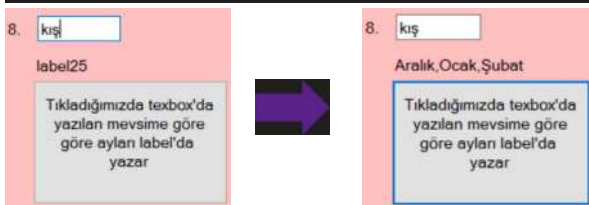
```
private void button6_Click(object sender, EventArgs e)
{
    int kitapadet;
    double toplam;
    kitapadet = Convert.ToInt32(textBox8.Text);
    if (kitapadet >= 0 && kitapadet <= 20)
    {
        toplam = (kitapadet * 8) - (kitapadet * 8 * 0.20);
        label17.Text = toplam + " TL";
    }
    if (kitapadet >= 21 && kitapadet <= 40)
    {
        toplam = (kitapadet * 8) - (kitapadet * 8 * 0.40);
        label17.Text = toplam + " TL";
    }
    if (kitapadet >= 41)
    {
        toplam = (kitapadet * 8) - (kitapadet * 8 * 0.50);
        label17.Text = toplam + " TL";
    }
}
```

23. Switch - Case Yapısı

- Switch yapısı şartların çok olduğu durumlarda yani dallanmanın fazla olduğu, sayıya göre ayı veren veya plakaya göre şehri yazan durumlarda kullanılır.
- Break komutunu case'in sonunda satırı burada bitir anlamında kullanılır.



```
private void button7_Click(object sender, EventArgs e)
{
    int ay = Convert.ToInt32(textBox9.Text);
    switch (ay)
    {
        case 1: label23.Text = "Ocak"; break;
        case 2: label23.Text = "Şubat"; break;
        case 3: label23.Text = "Mart"; break;
        case 4: label23.Text = "Nisan"; break;
        case 5: label23.Text = "Mayıs"; break;
        case 6: label23.Text = "Haziran"; break;
        case 7: label23.Text = "Temmuz"; break;
        case 8: label23.Text = "Ağustos"; break;
        case 9: label23.Text = "Eylül"; break;
        case 10: label23.Text = "Ekim"; break;
        case 11: label23.Text = "Kasım"; break;
        case 12: label23.Text = "Aralık"; break;
        default: label23.Text = "Hatalı Giriş"; break;
    }
}
```

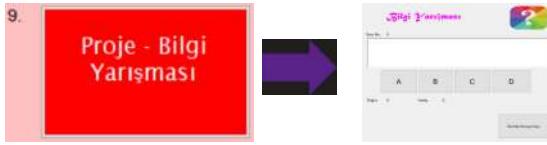


```
private void button8_Click(object sender, EventArgs e)
{
    string mevsim=textBox10.Text;
    switch (mevsim)
    {
        case "kış": label25.Text = "Aralık,Ocak,Şubat";
            break;
        case "ilkbahar": label25.Text = "Mart,Nisan,Mayıs";
            break;
        case "yaz": label25.Text = "Haziran,Temmuz,Ağustos";
            break;
        case "sonbahar": label25.Text = "Eylül,Ekim,Kasım";
            break;
        default: label25.Text = "Hatalı Giriş";
            break;
    }
}
```

24. Proje - Bilgi Yarışması / 1

25. Proje - Bilgi Yarışması / 2

- Butonun tıklanamaması için Enabled özelliğini kullanarak True yapıyoruz. Benzer işlemi pictureBox'ın görünürlüğünü kapatmak Visible'i false yapıyoruz..



```
private void button9_Click(object sender, EventArgs e)
{
    Form2 frm = new Form2();
    frm.Show();
}
```



```
int soruno = 0, dogru = 0, yanlis = 0;
```




```

private void btna_Click(object sender, EventArgs e)
{
    btna.Enabled = false;
    btnb.Enabled = false;
    btnc.Enabled = false;
    btnd.Enabled = false;
    btnsonraki.Enabled = true;

    label6.Text = btna.Text;
    if (label5.Text == label6.Text)
    {
        dogru++;
        lbldogru.Text = dogru.ToString();
        pictureBox2.Visible = true;
    }
    else
    {
        yanlis++;
        lblyanlis.Text = yanlis.ToString();
        pictureBox3.Visible = true;
    }
}

```

```

private void btnb_Click(object sender, EventArgs e)
{
    btna.Enabled = false;
    btnb.Enabled = false;
    btnc.Enabled = false;
    btnd.Enabled = false;
    btnsonraki.Enabled = true;

    label6.Text = btnb.Text;
    if (label5.Text == label6.Text)
    {
        dogru++;
        lbldogru.Text = dogru.ToString();
        pictureBox2.Visible = true;
    }
    else
    {
        yanlis++;
        lblyanlis.Text = yanlis.ToString();
        pictureBox3.Visible = true;
    }
}

```

```

private void btnc_Click(object sender, EventArgs e)
{
    btna.Enabled = false;
    btnb.Enabled = false;
    btnc.Enabled = false;
    btnd.Enabled = false;
    btnsonraki.Enabled = true;

    label6.Text = btnc.Text;
    if (label5.Text == label6.Text)
    {
        dogru++;
        lbldogru.Text = dogru.ToString();
        pictureBox2.Visible = true;
    }
    else
    {
        yanlis++;
        lblyanlis.Text = yanlis.ToString();
        pictureBox3.Visible = true;
    }
}

```



```

private void btnd_Click(object sender, EventArgs e)
{
    btna.Enabled = false;
    btnb.Enabled = false;
    btnc.Enabled = false;
    btnd.Enabled = false;
    btnsonraki.Enabled = true;

    label6.Text = btnd.Text;
    if (label5.Text == label6.Text)
    {
        dogru++;
        lbldogru.Text = dogru.ToString();
        pictureBox2.Visible = true;
    }
    else
    {
        yanlis++;
        lblyanlis.Text = yanlis.ToString();
        pictureBox3.Visible = true;
    }
}

```

```

private void btnsonraki_Click(object sender, EventArgs e)
{
    btna.Enabled = true;
    btnb.Enabled = true;
    btnc.Enabled = true;
    btnd.Enabled = true;
    btnsonraki.Enabled = false;

    pictureBox2.Visible = false;
    pictureBox3.Visible = false;

    //soruno++;
    soruno = soruno + 1;
    lblsoruno.Text = soruno.ToString();

    if (soruno == 1)
    {
        richTextBox1.Text = "Cumhuriyet kaç yılında ilan edilmiştir";
        btna.Text = "1920";
        btnb.Text = "1923";
        btnc.Text = "1915";
        btnd.Text = "1930";
        label5.Text = "1923";
    }
    if (soruno == 2)
    {
        richTextBox1.Text = "Hangi il Karadeniz'de bulunmaz";
        btna.Text = "Rize";
        btnb.Text = "Karabük";
        btnc.Text = "Bayburt";
        btnd.Text = "Çankırı";
        label5.Text = "Çankırı";
    }
    if (soruno == 3)
    {
        richTextBox1.Text = "İstiklal Marşı'nın yazarı kimdir?";
        btna.Text = "Mehmet Akif Ersoy";
        btnb.Text = "Mehmet Yozgatlı";
        btnc.Text = "Ömer Seyfettin";
        btnd.Text = "Reşat Nuri Gültekin";
    }
}

```

```
label5.Text = "Mehmet Akif Ersoy";
btnsonraki.Text = "Sonuçlar";
}
if (soruno == 4)
{
    btna.Enabled = false;
    btnb.Enabled = false;
    btnc.Enabled = false;
    btnd.Enabled = false;
    btnsonraki.Enabled = false;
    MessageBox.Show("Doğru:" + dogru + " \n " + "Yanlış:" + yanlis);
}
}
```

04 Döngüler

5 Ekim 2020 Pazartesi 13:36

04 Döngüler

26. Döngü Kavramı - Döngü Çeşitleri

- Döngüler tekrar gerektiren durumlarda kullanılır. Kullandığımız döngüler for, while, do/while, foreach'dir. Foreach dizilerde için kullanılır.

27. For Döngüsü

- For döngüsünün üç parametresi vardır ve bunlar noktalı virgül ile ayrılırlar. 1.Başlangıç, 2.Bitiş ve 3.Artış veya Azalış'dır.

1. listBox1

Tıkladığımızda belirlediğimiz döngüye göre yazdığımız metni listBox'a ekler

→

1. 1. Merhaba
2. Merhaba
3. Merhaba
4. Merhaba
5. Merhaba

Tıkladığımızda belirlediğimiz döngüye göre yazdığımız metni listBox'a ekler

```
private void button1_Click(object sender, EventArgs e)
{
    int i;
    for (i = 1; i <=5; i++)
    {
        listBox1.Items.Add(i + ". " + "Merhaba");
    }
}
```

2. listBox2

Tıkladığımızda belirlediğimiz döngüye göre yazdığımız metni listBox'a ekler

→

2. 1
3
5
7
9

Tıkladığımızda belirlediğimiz döngüye göre yazdığımız metni listBox'a ekler

28. Break Point Kullanımı

- Uygulamamızı istediğimiz satırda duraklatıp, satır satır devam ederek tüm süreci takip edebilmemizi sağlayan yapıdır.

29. Pratik Örnek - Ardışık Sayı Toplamları

3. label4

Tıkladığımızda belirlediğimiz döngüye göre ardışık toplamı label'a yazar

→

3. 55

Tıkladığımızda belirlediğimiz döngüye göre ardışık toplamı label'a yazar

```
private void button3_Click(object sender, EventArgs e)
{
    int toplam = 0;
    for (int i = 1; i <= 10; i++)
    {
        toplam = toplam + i;
    }
    label4.Text = toplam.ToString();
}
```

4. label5

Tıkladığımızda belirlediğimiz döngüye göre 1 ile 10 arasındaki çift sayıların toplamını label'a yazar.

→

4. 30

Tıkladığımızda belirlediğimiz döngüye göre 1 ile 10 arasındaki çift sayıların toplamını label'a yazar.

```
private void button4_Click(object sender, EventArgs e)
{
    int sayi = 0;
    for (int i = 2; i <= 10; i=i+2)
    {
        sayi = sayi + i;
    }
    label5.Text = sayi.ToString();
}
```

30. Karar Yapıları ile Döngü Kullanımı

5. listBox3

Tıkladığımızda belirlediğimiz döngüye göre 1 ile 100 arasındaki 5 ve 7'ye tam bölünen sayıları listBox'a yazar.

5. 35
70

Tıkladığımızda belirlediğimiz döngüye göre 1 ile 100 arasındaki 5 ve 7'ye tam bölünen sayıları listBox'a yazar.

```
private void button5_Click(object sender, EventArgs e)
{
    for (int i = 1; i <=100; i++)
    {
        if (i%5==0 && i%7==0)
        {
            listBox3.Items.Add(i);
        }
    }
}
```

6. Sayı: 12

Tıkladığımızda textbox'da girilen sayıya kadar olan sayıların tam bölünenleri listBox'a yazar

6. Sayı: 12

1
2
3
4
6
12

Tıkladığımızda textbox'da girilen sayıya kadar olan sayıların tam bölünenleri listBox'a yazar

```
private void button6_Click(object sender, EventArgs e)
{
    int a = Convert.ToInt32(textBox1.Text);
    for (int i = 1; i <= a; i++)
    {
        if (a % i == 0)
        {
            listBox4.Items.Add(i);
        }
    }
}
```

31. While Döngüsü

- While döngüsü şartı sağladıkça işlemi yapmaya devam eder.

7. listBox5

listBox6

Tıkladığımızda while döngüsü ile ilgili iki örneğimizin birincisi listBox'da ikincisi mesaj kutusu ile çıkar

7. Merhaba Rize
Merhaba Rize

1
2

Tıkladığımızda while döngüsü ile ilgili iki örneğimizin birincisi listBox'da ikincisi mesaj kutusu ile çıkar

15

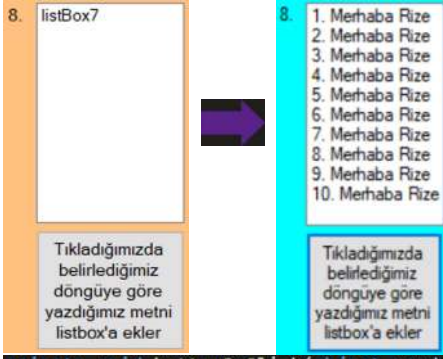
Tamam

```
private void button7_Click(object sender, EventArgs e)
{
    int i = 1;
    while (i<=2)
    {
        listBox5.Items.Add("Merhaba Rize");
        listBox6.Items.Add(i);
        i++;
    }
    int j = 1;
    int toplam=0;
    while (j<=5)
    {
        toplam = toplam + j;
        j++;
    }
    MessageBox.Show(toplam.ToString());
}
```

32. Do While Döngüsü

- Do/while döngüsünün while'den tek farkı ilk önce işlemi yapıyor sonra while döngüsüne geçiyor. While'de ise şart

sağlanmadan işlemi yapmıyor.



```
private void button8_Click(object sender, EventArgs e)
{
    int i = 1;
    do
    {
        listBox7.Items.Add(i+ ". " + "Merhaba Rize");
        i++;
    }
    while (i<=10);
}
```

33. Timer Kullanımı

- Enabled, timer nesnesinin çalışıp çalışmayacağını belirten özelliktir. Bu değer True ise timer çalışır anlamında, false ise timer çalışmaz anlamına gelmektedir.
- Interval, timer'ın milisaniye cinsinden çalıştığını belirttiğimiz yerdir. 1000ms=1s
- Tick fonksiyonu timer'ın intervalde belirtilen zamanda yapılmasını istediğimiz komutları yazdığımız yerdir.
- This komutu formumuzu kasteder. Çünkü form bir sınıf olduğu için kodda nesne olarak çağırıştırmamız gerekir.

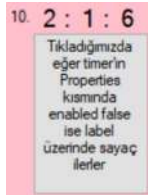


```
private void button9_Click(object sender, EventArgs e)
{
    timer1.Start();
}

int sayac = 0;
private void timer1_Tick(object sender, EventArgs e)
{
    sayac++;
    label13.Text = sayac.ToString();
    if (sayac==20)
    {
        //timer1.Stop();
        //label13.Text = "Bitti";
        this.BackColor = Color.Aqua;
    }
    if (sayac==40)
    {
        this.BackColor = Color.Pink;
    }
    if(sayac==60)
    {
        this.BackColor = Color.Yellow;
        sayac = 0;
    }
}
```

34. Pratik Örnek - Timer ile Dijital Saat

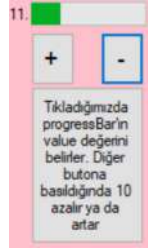
- Timer'ın enabled özelliğini true yaparsak eğer f5'e basıldığı an button'a tıklamaya gerek kalmadan çalışır.



```
private void button10_Click(object sender, EventArgs e)
{
    //timer2.Start();
}

int saat = 0, dakika = 0, saniye = 0;
private void timer2_Tick(object sender, EventArgs e)
{
    saniye++;
    label17.Text = saniye.ToString();
    if (saniye==60)
    {
        dakika++;
        label16.Text = dakika.ToString();
        saniye = 0;
        if (dakika == 3)
        {
            saat++;
            label15.Text = saat.ToString();
            dakika = 0;
        }
    }
}
```

35. ProgressBar Kullanımı

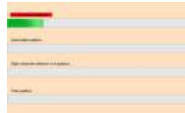


```
private void button13_Click(object sender, EventArgs e)
{
    progressBar1.Value = 75;
}

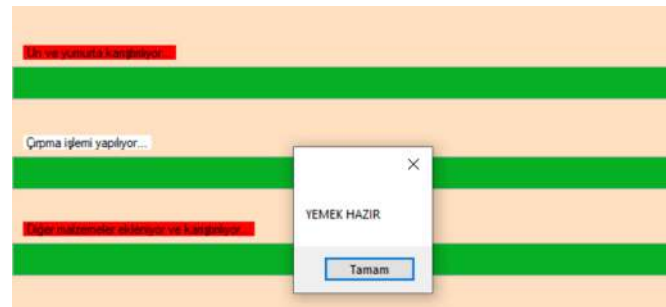
private void button11_Click(object sender, EventArgs e)
{
    progressBar1.Value += 10;
}

private void button12_Click(object sender, EventArgs e)
{
    progressBar1.Value -= 10;
}
```

36. Proje - Pasta Pişirme Süreç Uygulaması



```
private void button14_Click(object sender, EventArgs e)
{
    Form2 frm = new Form2();
    frm.Show();
}
```



Diğer malzemeler ekleniyor ve karıştırılıyor...

Pasta pişiriliyor...

```
private void timer1_Tick(object sender, EventArgs e)
{
    progressBar1.Value += 10;

    if (progressBar1.Value % 20 == 10)
    {
        label1.BackColor = Color.White;
    }
    if (progressBar1.Value % 20 == 0)
    {
        label1.BackColor = Color.Red;
    }
    if (progressBar1.Value == 100)
    {
        timer1.Stop();
        timer2.Start();
    }
}

private void timer2_Tick(object sender, EventArgs e)
{
    progressBar2.Value += 5;

    if (progressBar2.Value % 10 == 0)
    {
        label2.BackColor = Color.White;
    }
    if (progressBar2.Value % 10 == 5)
    {
        label2.BackColor = Color.Red;
    }
    if (progressBar2.Value == 100)
    {
        timer2.Stop();
        timer3.Start();
    }
}

private void timer3_Tick(object sender, EventArgs e)
{
    progressBar3.Value += 20;

    if (progressBar2.Value % 40 == 0)
    {
        label3.BackColor = Color.White;
    }
    if (progressBar3.Value % 40 == 20)
    {
        label3.BackColor = Color.Red;
    }
    if (progressBar3.Value == 100)
    {
        timer3.Stop();
        timer4.Start();
    }
}

private void timer4_Tick(object sender, EventArgs e)
{
    progressBar4.Value += 2;

    if (progressBar4.Value % 2 == 0)
    {
```

Diğer malzemeler ekleniyor ve karıştırılıyor...

timer1_Tick

Tamam

Pasta pişiriliyor...

```
private void timer4_Tick(object sender, EventArgs e)
{
    progressBar4.Value += 2;

    if (progressBar4.Value % 2 == 0)
    {
        label4.BackColor = Color.White;
    }
    if (progressBar4.Value % 2 == 1)
    {
        label4.BackColor = Color.Red;
    }
    if (progressBar4.Value == 100)
    {
        timer4.Stop();
        MessageBox.Show("YEMEK HAZIR");
    }
}
```

05 Diziler

7 Ekim 2020 Çarşamba 17:27

05 Diziler

37. Dizi Nedir - Teorik Bilgiler

- Dizilere örnek verilirse, Renkler: mavi,sarı,kırmızı ; Futbol Takımları: Fenerbahçe,Galatasaray gibi. Diziler bellekte geçici olan ram'de tutulur. Index olarak sıralaması vardır ve 0'dan başlar. Gösterimi ise [] şeklindedir.

38. Dizi Tanımlama ve Eleman Yazdırma

- Dizilerin söz dizimi değişkenti [] diziadı = { };



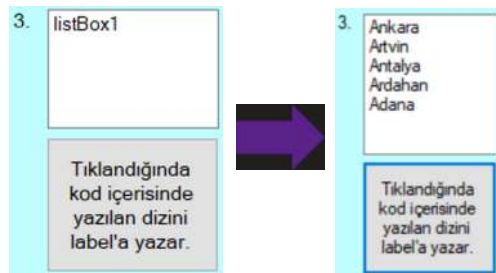
```
private void button1_Click(object sender, EventArgs e)
{
    string [] kisiler = { "Ali", "Mehmet", "Buket", "Osman", "Cengiz", "Yaren", "Betül" };
    label2.Text = kisiler[5];
}
```



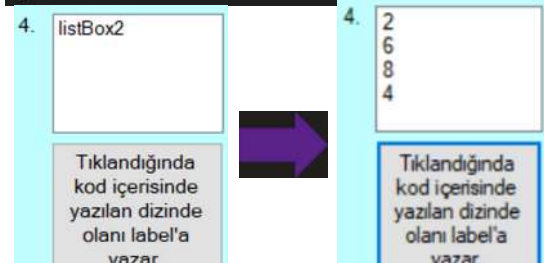
```
private void button2_Click(object sender, EventArgs e)
{
    int[] sayılar = {4,5,7,5,52,04,7};
    label3.Text = sayılar[4].ToString();
}
```

39. Dizi Elemanları Listeleme

- Length ile index farklı şeylerdir. Index sıfırdan başlarken, Length eleman sayısını verir.



```
private void button3_Click(object sender, EventArgs e)
{
    string[] sehirler = { "Ankara", "Artvin", "Antalya", "Ardahan", "Adana" };
    for (int i = 0; i < sehirler.Length; i++)
    {
        listBox1.Items.Add(sehirler[i]);
    }
}
```



kod içerisinde yazılan dizinde olanı label'a yazar.


kod içerisinde yazılan dizinde olanı label'a yazar.

```
private void button4_Click(object sender, EventArgs e)
{
    int [] sayılar = { 2, 3, 5, 6, 8, 4, 1, 9};
    for (int i = 0; i < sayılar.Length; i++)
    {
        if (sayılar[i]%2==0)
        {
            listBox2.Items.Add(sayılar[i]);
        }
    }
}
```

40. Foreach Döngüsü

- Foreach döngüsü 4 parametreden oluşur. 1.değişken türü, 2.değişken adı, 3.in(içinden), 4.Dizi adı
- Dizilerde for döngüsü kullanarak listbox'a yazdırdığımızı foreach kullanarak daha kolay bir şekilde yazdırabiliyoruz.

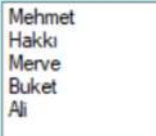
5. listBox3



Tıklandığında kod içerisinde yazılan dizinde olanı foreach kullanarak label'a yazar



5. Mehmet
Hakkı
Merve
Buket
Ali



Tıklandığında kod içerisinde yazılan dizinde olanı foreach kullanarak label'a yazar

```
private void button5_Click(object sender, EventArgs e)
{
    string [] rehber = { "Mehmet", "Hakkı", "Merve", "Buket", "Ali" };
    foreach (string a in rehber)
    {
        listBox3.Items.Add(a);
    }
}
```

6. listBox4

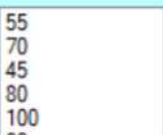


Toplam: label11
Ortalama: label12

Tıklandığında kod içerisinde yazılan dizinde olanı foreach kullanarak



6. 55
70
45
80
100
60
60
70
90



Toplam: 630
Ortalama: 70

Tıklandığında kod içerisinde yazılan dizinde olanı foreach kullanarak

```
private void button6_Click(object sender, EventArgs e)
{
    int toplam = 0;
    int [] sayilarimiz = { 55,70,45,80,100,60,60,70,90 };
    foreach (int a in sayilarimiz)
    {
        listBox4.Items.Add(a);
        toplam = toplam + a;
        label11.Text = toplam.ToString();
    }
    int ortalama = toplam / sayilarimiz.Length;
    label12.Text = ortalama.ToString();
}
```

06 Metotlar

8 Ekim 2020 Perşembe

12:11

06 Metotlar

41. Metot Nedir - Teorik Bilgi

- Metot; komutları bir araya toplayan yapılardır. Tekrar eden durumları tek bir küme haline getirerek aynı şeyleri bir daha yazmadan kümeyi çağırarak daha az komut satırı kullanılmış olur.

42. Geriye Değer Döndürmeyen Metotlar

- Metotun kümesi için private ya da public kullanılır. Eğer kullanmassak private olarak algılar. Private içindeki bilgiyi sadece o form için kullanılır onun dışında kullanılamaz. void geriye değer döndürmeyen'dir.
- Metodu global dışında oluşturduktan sonra buttonda yazarken küp işareti çıkar ve altındaki anahtar ise private olduğunu gösterir.



```
private void temizle()
{
    textBox1.Text = "";
    textBox2.Text = "";
    textBox3.Text = "";
    textBox4.Text = "";
    textBox1.Focus();
}
void renklendir()
{
    textBox1.BackColor = Color.Aqua;
    textBox2.BackColor = Color.Yellow;
    textBox3.BackColor = Color.Peru;
    textBox4.BackColor = Color.Red;
}
```

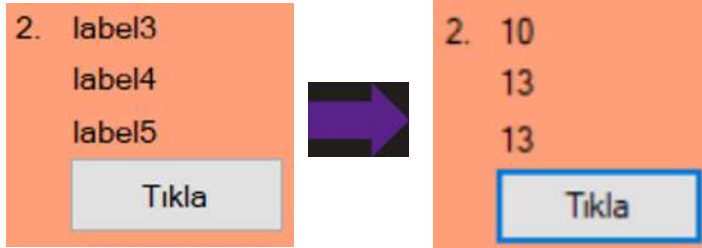
```
private void button1_Click(object sender, EventArgs e)
{
    temizle();
}
```



```
private void button2_Click(object sender, EventArgs e)
{
    renklendir();
}
```

43. Geriye Değer Döndüren Metotlar

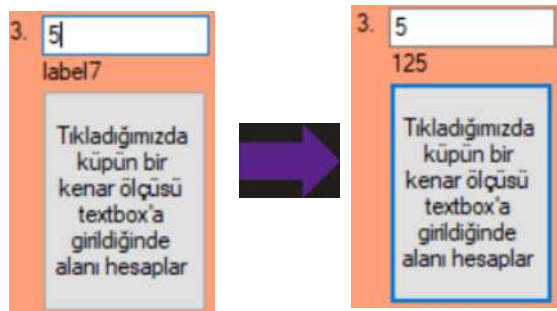
- Geriye değer döndürenlerde de void yerine int, string gibi değişkenleri kullanırız.
- Metot için verdiğimiz adın parantez içindekileri metotun parametreleri sayılır ve her değişken için değişken türünü yazmamız gerekir.



```
int Toplam(int s1, int s2)
{
    int s3 = s1 + s2;
    return s3;
}

private void button3_Click(object sender, EventArgs e)
{
    label3.Text = Toplam(4, 6).ToString();
    label4.Text = Toplam(7, 6).ToString();
    label5.Text = Toplam(4, 9).ToString();
}
```

44. Pratik Örnek - Küp Hesaplayan Metot



```
int Kup(int say1)
{
    int sonuc = say1 * say1 * say1;
    return sonuc;
}

private void button4_Click(object sender, EventArgs e)
{
    int s = Convert.ToInt32(textBox5.Text);
    label7.Text = Kup(s).ToString();
}
```

07 Sınıflar

8 Ekim 2020 Perşembe

17:37

07 Sınıflar

45. Sınıf Nedir - Teorik Bilgi

- Sınıflar kod tekrarını azaltma ve gizlilik konularında kullanılabilir. Sınıf kullanırken önce sınıfı belirlemeli örnek verirken sonra nesneyi belirleyerek sınıf ile özellikler arasında bağ kurarız, bu da odalar olabilir. Daha sonra nesnenin özelliklerini belirleriz. Bunlar renk, fiyat, oda tipi gibi olabilirler.

46. Sınıf Oluşturma ve Nesne Kavramı

- Değişkenlerin erişim türünü public belirleyerek diğer yerlerden yani formlardan da erişmesini sağlarız.
- Form'un kod kısmında önce Araba sınıfından nesne türetiriz. Sınıfın ismi olan Arabayı çağırdıktan sonra nesne olarak rb isminde nesne üretiyoruz.
- Kod olmadan form geçişini solution bölümündeki program kısmına çift tıklarız ve kod kısmındaki form1 yerine form2'yi yazarız. Aslında program çalıştığında bize form2'yi göster demektir.

```
class Araba
{
    public string renk;
    public int hız;
    public double motorgucu;
    public char durum;
    public int fiyat;
}
```

1. label2		1. Mavi
label3		200
label4	➡	1245,56
label5		160000
label6		s

```
private void Form1_Load(object sender, EventArgs e)
{
    Araba rb = new Araba();
    rb.renk = "Mavi";
    rb.hız = 200;
    rb.motorgucu = 1245.56;
    rb.fiyat = 160000;
    rb.durum = 's';
    label2.Text = rb.renk;
    label3.Text = rb.hız.ToString();
    label4.Text = rb.motorgucu.ToString();
    label5.Text = rb.fiyat.ToString();
    label6.Text = rb.durum.ToString();
}
```

1. label2		1. Yellow
label3		220
label4	➡	1485,95
label5		180000

label4	1485,95
label5	180000
label6	s

```
private void Form2_Load(object sender, EventArgs e)
{
    Araba rb2 = new Araba();
    rb2.renk = "Yellow";
    rb2.hız = 220;
    rb2.motorgucu = 1485.95;
    rb2.fiyat = 180000;
    rb2.durum = 's';
    label2.Text = rb2.renk;
    label3.Text = rb2.hız.ToString();
    label4.Text = rb2.motorgucu.ToString();
    label5.Text = rb2.fiyat.ToString();
    label6.Text = rb2.durum.ToString();
}
```

47. Kapsülleme

- Kapsülleme, private tanımlanan nesnenin metotları, verileri ve değişkenleri gibi alanları public hale getirerek diğer nesnelerden saklayarak ve erişimi sınırlandırarak bilinçsiz kullanımı önleyerek bizim istediğimiz formatta kullanmamızı sağlayan yapıdır.
- Public kullanımında erişim sağlanıyordu ama private kullanınca erişilemiyordu. Private kullanımında erişilebilmesi için alta başka formatta public kullanarak erişilebilirlik sağlanır. Bu kullanımda get ve set blokları kullanılır. Get ile geriye döndürecek değeri yazarız. Bunu ayarlamak için de set kullanırız.
- Math.Abs kullanımı sayının mutlak değerini alır.
- ToUpper kullanımı tüm harfleri büyük harflerle yazdırır.

```

class Araba
{
    public string renk;
    public int hız;
    public double motorgucu;
    public char durum;
    public int fiyat;
    private int yıl;
    private string marka;

    2 references
    public int YIL
    {
        get { return yıl; }
        set { yıl = Math.Abs(value); }
    }

    2 references
    public string MARKASI
    {
        get { return marka; }
        set { marka = value.ToUpper(); }
    }
}

```

2. label7
label8



2. 2016
GOLF

```

private void Form1_Load(object sender, EventArgs e)
{
    Araba rb = new Araba();
    rb.renk = "Mavi";
    rb.hız = 200;
    rb.motorgucu = 1245.56;
    rb.fiyat = 160000;
    rb.durum = 's';
    rb.YIL = -2016;
    rb.MARKASI = "Golf";
    label2.Text = rb.renk;
    label3.Text = rb.hız.ToString();
    label4.Text = rb.motorgucu.ToString();
    label5.Text = rb.fiyat.ToString();
    label6.Text = rb.durum.ToString();
    label7.Text = rb.YIL.ToString();
    label8.Text = rb.MARKASI;
}

```

48. Kalıtım

- class kısmındaki Arabaya : koyup sonra diğer sınıfın ismini yazarsa Arabada'ki değerleri ArabaDetay kısmına eklemiş olarak kalıtım yapmış olduk.


```
class ArabaDetay
{
    public string plaka;
    public int muayene;
    public string sahip;
}
class Araba : ArabaDetay
```

3. label9
label10
label11



3. 2017
34 AİM 1981
Zübeyde Hanım

```
private void Form1_Load(object sender, EventArgs e)
{
    Araba rb = new Araba();
    rb.renk = "Mavi";
    rb.hız = 200;
    rb.motorgucu = 1245.56;
    rb.fiyat = 160000;
    rb.durum = 's';
    rb.YIL = -2016;
    rb.MARKASI = "Golf";
    rb.muayene = 2017;
    rb.plaka = "34 AİM 1981";
    rb.sahip = "Zübeyde Hanım";
    label2.Text = rb.renk;
    label3.Text = rb.hız.ToString();
    label4.Text = rb.motorgucu.ToString();
    label5.Text = rb.fiyat.ToString();
    label6.Text = rb.durum.ToString();
    label7.Text = rb.YIL.ToString();
    label8.Text = rb.MARKASI;
    label9.Text = rb.muayene.ToString();
    label10.Text = rb.plaka;
    label11.Text = rb.sahip;
}
```

49. Console Ekranı

- Console yani siyah ekranın hep kalması için read veya readKey kullanılabilir. ReadKey kullanımda siyah ekrana klavyeden bir kez tuşlayarak yazabiliyoruz.
- WriteLine komutu kendisinden sonraki Write komutunu bir alt satıra yazar.

Merhaba
Dünya

```
Console.WriteLine("Merhaba");
Console.Write("Dünya");
Console.Read();
```

```
29 Ekim
5
```

```
string kelime;
kelime = "29 Ekim";
Console.WriteLine(kelime);
int sayi;
sayi = 5;
Console.Write(sayi);
Console.ReadKey();
```

50. Sınıf İçinde Geriye Değer Döndürmeyen Metotlar / 1

```
class Mesaj
{
    1 reference
    public void metin()
    {
        Console.Write("Merhaba Karabük");
    }
}
```

```
Merhaba Karabük
```

```
Mesaj msj = new Mesaj();
msj.metin();
Console.Read();
```

51. Sınıf İçinde Geriye Değer Döndürmeyen Metotlar / 2

- ReadLine komutu klavyeden veri okumak için kullanılır.

```
class Kisiler
{
    1 reference
    public void kisilistesi()
    {
        for (int i = 0; i < 10; i++)
        {
            Console.WriteLine(i + ". Kişi");
        }
    }
}
```

```
0. Kişi
1. Kişi
2. Kişi
3. Kişi
4. Kişi
5. Kişi
6. Kişi
7. Kişi
8. Kişi
```



```
0. Kişi  
1. Kişi  
2. Kişi  
3. Kişi  
4. Kişi  
5. Kişi  
6. Kişi  
7. Kişi  
8. Kişi  
9. Kişi
```

```
Kisiler ks = new Kisiler();  
ks.kisilistesi();  
Console.Read();
```

İsminizi Girin: _



İsminizi Girin: Cengizhan Topçu

```
string ads;  
Console.Write("İsminizi Girin: ");  
ads = Console.ReadLine();
```

```
class Kisiler  
{  
    //public void kisilistesi()  
    //{  
    //    for (int i = 0; i < 10; i++)  
    //    {  
    //        Console.WriteLine(i + ". Kişi");  
    //    }  
    //}  
  
    0 references  
    public void rehberlist(string adsoyad)  
    {  
        for (int i = 0; i < 10; i++)  
        {  
            Console.WriteLine(adsoyad);  
        }  
    }  
}
```

```
Cengizhan Topçu
Cengizhan Topçu
Cengizhan Topçu
Cengizhan Topçu
Cengizhan Topçu
Cengizhan Topçu
Cengizhan Topçu
Cengizhan Topçu
Cengizhan Topçu
Cengizhan Topçu
Tuncel Akbaba
Tuncel Akbaba
Tuncel Akbaba
Tuncel Akbaba
Tuncel Akbaba
Tuncel Akbaba
Tuncel Akbaba
Tuncel Akbaba
Tuncel Akbaba
Tuncel Akbaba
İsminizi Girin: Mahmut Kaşgarlı
Mahmut Kaşgarlı
Mahmut Kaşgarlı
Mahmut Kaşgarlı
Mahmut Kaşgarlı
Mahmut Kaşgarlı
Mahmut Kaşgarlı
Mahmut Kaşgarlı
Mahmut Kaşgarlı
Mahmut Kaşgarlı
Mahmut Kaşgarlı
```

```
string ads;
Console.Write("İsminizi Girin: ");
ads = Console.ReadLine();
Kisiler rl = new Kisiler();
rl.rehberlist("Cengizhan Topçu");
rl.rehberlist("Tuncel Akbaba");
rl.rehberlist(ads);
Console.Read();
```

52. Sınıf İçinde Geriye Değer Döndüren Metotlar

```
class işlem
{
    2 references
    public int toplama(int s1, int s2)
    {
        int s3 = s1 + s2;
        Console.WriteLine("Sonuç: " + s3);
        return s3;
    }
    1 reference
    public int kare(int deger)
    {
        int sonuc = deger * deger;
        Console.WriteLine("Sonuç: " + sonuc);
        return sonuc;
    }
}
Sonuc: 11
```

```
Sonuç: 11
Sonuç: 22

Sonuç: 25
```

```
islem isl = new islem();
isl.topla(4, 7);
isl.topla(5, 17);
Console.WriteLine("\n\n\n");
isl.kare(5);
Console.Read();
```

53. Constructor (Yapıcı) Metotlar / 1

- Yapıcı, birinci olarak sınıftaki özelliklere ilk değer atamada kullanılır. İkinci olarak sınıftan bir nesne oluşturulduğu an çalıştırılmak istenilen durumlarda kullanılır. Üçüncü olarak sınıf adı ile yapıcı adı aynı olmak zorunda.

```
class Ogrenci
{
    0 references
    public Ogrenci()
    {
        Console.WriteLine("Ad: Cengizhan");
        Console.WriteLine("Soyad: Topçu");
        Console.WriteLine("Meslek: Memur");
    }
}
```

```
Ad: Cengizhan
Soyad: Topçu
Meslek: Memur
```

```
Ogrenci ogr = new Ogrenci();
public Ogrenci(string bilgi)
{
    Console.WriteLine("Durum: " + bilgi);
}
```

```
Durum: Cengizhan-Kızılay
```

```
Ogrenci ogr = new Ogrenci("Cengizhan-Kızılay");
Console.Read();
public Ogrenci(string bilgi)
{
    Console.WriteLine("Durum: " + bilgi);
}
```

```
Ad-Kulüp Adı:Cengizhan-Kızılay
```

```
Ad-Kulüp Adı:Cengizhan-Kızılay  
Durum: Cengizhan-Kızılay
```

```
string blg;  
Console.Write("Ad-Kulüp Adı:");  
blg = Console.ReadLine();  
Ogrenci ogr = new Ogrenci(blg);  
Console.Read();
```

54. Constructor (Yapıcı) Metotlar / 2

- ToLower kullanımı tüm harfleri küçük harflerle yazdırır.
- Yapıcılar için ilk değer olayında eğer ilk değeri vermez isek sınıf içerisindeki ilk değeri alır. Program kısmına yazarsam onu yazar.

```

class Kimlik
{
    string ad;
    string soyad;
    string memleket;
    int yas;
    char cinsiyet;

    2 references
    public string AD
    {
        get { return ad; }
        set { ad = value.ToLower(); }
    }

    2 references
    public string SOYAD
    {
        get { return soyad; }
        set { soyad = value.ToUpper(); }
    }

    2 references
    public string MEMLEKET
    {
        get { return memleket; }
        set { memleket = value; }
    }

    1 reference
    public int YAS
    {
        get { return yas; }
        set { yas = Math.Abs(value); }
    }

    0 references
    public char CINSIYET
    {
        get { return cinsiyet; }
        set { cinsiyet = value; }
    }

    1 reference
    public Kimlik()
    {
        ad = "";
        soyad = "";
        yas = 18;
        cinsiyet = 'K';
        memleket = "Ankara";
    }
}

```

cengizhan
TOPÇU

cengizhan
TOPÇU
18
Rize

```
Kimlik kml = new Kimlik();  
kml.AD = "Cengizhan";  
kml.SOYAD = "Topçu";  
kml.MEMLEKET = "Rize";  
Console.WriteLine(kml.AD);  
Console.WriteLine(kml.SOYAD);  
Console.WriteLine(kml.YAS);  
Console.WriteLine(kml.MEMLEKET);  
Console.Read();
```

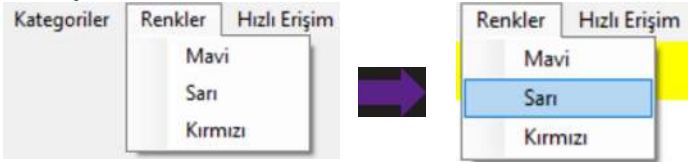

08 Ekstra Araçlar

9 Ekim 2020 Cuma 16:40

08 Ekstra Araçlar

55. Menu Strip Kullanımı

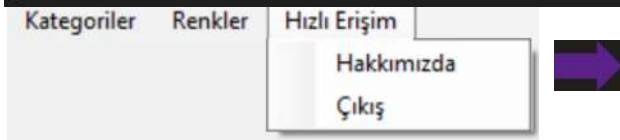
- MessageBox'ın 4 parametresi vardır. Birincisi metnin içeriği, ikincisi başlık, üçüncüsü buton, dördüncüsü icon'dur.



```
private void maviToolStripMenuItem_Click(object sender, EventArgs e)
{
    this.BackColor = Color.Aqua;
}
```

```
private void sarıToolStripMenuItem_Click(object sender, EventArgs e)
{
    this.BackColor = Color.Yellow;
}
```

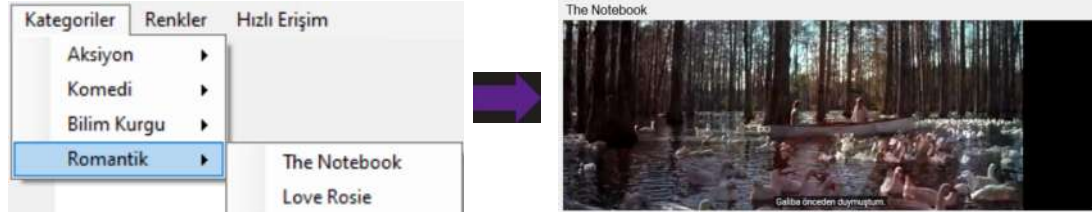
```
private void kırmızıToolStripMenuItem_Click(object sender, EventArgs e)
{
    this.BackColor = Color.Red;
}
```



```
private void hakkımızdaToolStripMenuItem_Click(object sender, EventArgs e)
{
    MessageBox.Show("Bu uygulama Maker.Com tarafından yapılmıştır", "bilgi", MessageBoxButtons.OK, MessageBoxIcon.Information);
}
```

```
private void çıkışToolStripMenuItem_Click(object sender, EventArgs e)
{
    Application.Exit();
}
```

56. Web Browser Kullanımı / Sinema Fragman Projesi Hazırlama /2



```

private void johnWickToolStripMenuItem_Click(object sender, EventArgs e)
{
    webBrowser1.Navigate("https://www.youtube.com/watch?v=C0BMx-qxsP4");
    label1.Text = "John Wick";
}

1 reference
private void deadpoolToolStripMenuItem_Click(object sender, EventArgs e)
{
    webBrowser1.Navigate("https://www.youtube.com/watch?v=jCXmDgYhkH0");
    label1.Text = "Deadpool";
}

1 reference
private void avatarToolStripMenuItem_Click(object sender, EventArgs e)
{
    webBrowser1.Navigate("https://www.youtube.com/watch?v=aVd0-cx-McA");
    label1.Text = "Avatar";
}

1 reference
private void yildizlararasıToolStripMenuItem_Click(object sender, EventArgs e)
{
    webBrowser1.Navigate("https://www.youtube.com/watch?v=vVJeYMRam0o");
    label1.Text = "Yıldızlararası";
}

1 reference
private void theNotebookToolStripMenuItem_Click(object sender, EventArgs e)
{
    webBrowser1.Navigate("https://www.youtube.com/watch?v=OCXwBp3vEZ8");
    label1.Text = "The Notebook";
}

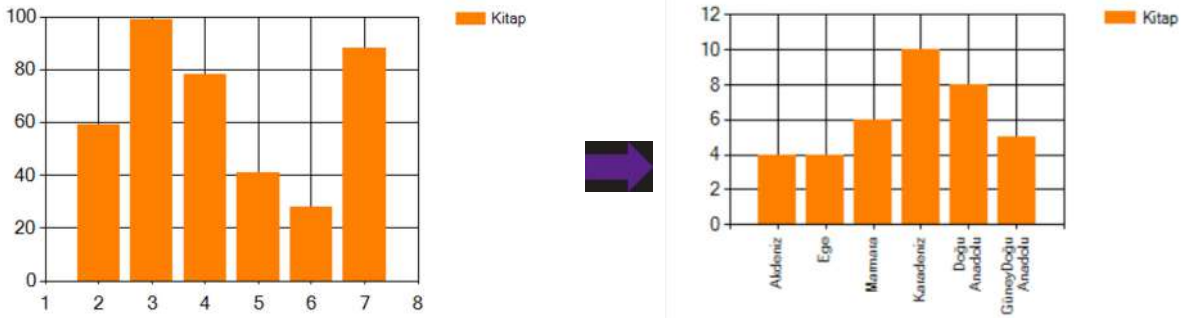
1 reference
private void loveRosieToolStripMenuItem_Click(object sender, EventArgs e)
{
    webBrowser1.Navigate("https://www.youtube.com/watch?v=5zL3YJKygd4");
    label1.Text = "Love Rosie";
}

```

57. Chart (Grafik) Aracı Kullanımı

Chart'ın Dock özelliğini ortaya basıp Fill yaparsam form büyütüp ya da küçülttüğümüzde grafik'te aynı şekilde büyür ya da küçülür. Grafiğin ile oynamak için Series'a tıklar tipini ChartType'dan, ismini Data kısmındaki Name'den değiştiririz. Rengini değiştirmek için palette kullanılır.

- Grafiğe bilgilerimizi eklerken grafiğe verilen ismi koda köşeli parantez içinde kullanırız. Pointsleri kullanarak lokasyonları yani x ve y düzlemlerin bilgilerini tutar.



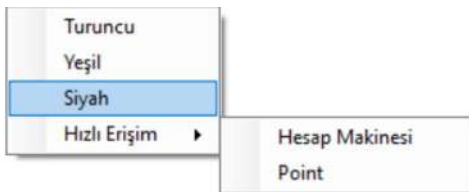
```

private void Form1_Load(object sender, EventArgs e)
{
    chart1.Series["Kitap"].Points.AddXY("Akdeniz",4);
    chart1.Series["Kitap"].Points.AddXY("Ege",4);
    chart1.Series["Kitap"].Points.AddXY("Marmara",6);
    chart1.Series["Kitap"].Points.AddXY("Karadeniz",10);
    chart1.Series["Kitap"].Points.AddXY("Doğu Anadolu",8);
    chart1.Series["Kitap"].Points.AddXY("GüneyDoğu Anadolu",5);
}

```

58. Context Menu Strip Kullanımı

- Context Menu aracı fare ile sağ tık yaptığımızda gelen kutucuktur. F5 yaptığımızda da aynı görevi görecek. Menüü yaptıktan sonra çalışabilmesi için ayarlarından ContextMenüStrip'den araç seçilmelidir.



```
private void turuncuToolStripMenuItem_Click(object sender, EventArgs e)
{
    this.BackColor = Color.Orange;
}

1 reference
private void siyahToolStripMenuItem_Click(object sender, EventArgs e)
{
    this.BackColor = Color.Black;
}

1 reference
private void yeşilToolStripMenuItem_Click(object sender, EventArgs e)
{
    this.BackColor = Color.Green;
}

1 reference
private void hesapMakinesiToolStripMenuItem_Click(object sender, EventArgs e)
{
    System.Diagnostics.Process.Start("calc.exe");
}

1 reference
private void pointToolStripMenuItem_Click(object sender, EventArgs e)
{
    System.Diagnostics.Process.Start("mspaint.exe");
}
```

09 Ekstra Yapılar

9 Ekim 2020 Cuma 20:45

09 Ekstra Yapılar

59. Random Komutu

- Random, rastgele sayı atma sınıfıdır.
- Next ile yazdığımızda ilk değer dahil ikinci değer dahil değil.

1. label2

Tıklandığında
labelde rasgele
sayı yazar.

1. 3

Tıklandığında
labelde rasgele
sayı yazar.

```
private void button1_Click(object sender, EventArgs e)
{
    Random rnd = new Random();
    int sayi;
    sayi = rnd.Next(1, 10);
    label2.Text = sayi.ToString();
}
```

2. label3
label5
label6

Tıklandığında
labelde rasgele
sayı yazar.

2. 3
77
646

Tıklandığında
labelde rasgele
sayı yazar.

```
private void button2_Click(object sender, EventArgs e)
{
    Random rd = new Random();
    int s1, s2, s3;
    s1 = rd.Next(1, 9);
    label3.Text = s1.ToString();
    s2 = rd.Next(10, 99);
    label5.Text = s2.ToString();
    s3 = rd.Next(100, 999);
    label6.Text = s3.ToString();
}
```

60. Örnek Proje - Sayısal Loto Uygulaması

3. label7
label8
label9
label11

Tıklandığında
labelde rasgele
sayı yazar ve
textbox'da
yazılanın doğru
ve yanlış
olmasına göre
arka rengi değişir

3. 1
3
2
3

1
4
3
1

Tıklandığında
labelde rasgele
sayı yazar ve
textbox'da
yazılanın doğru
ve yanlış
olmasına göre
arka rengi değişir


```

private void button3_Click(object sender, EventArgs e)
{
    Random loto = new Random();
    int say1, say2, say3, say4;
    say1 = loto.Next(1,4);
    say2 = loto.Next(1, 4);
    say3 = loto.Next(1, 4);
    say4 = loto.Next(1, 4);
    label7.Text = say1.ToString();
    label8.Text = say2.ToString();
    label9.Text = say3.ToString();
    label11.Text = say4.ToString();

    if (label7.Text==textBox1.Text)
    {
        textBox1.BackColor = Color.Green;
    }
    else
    {
        textBox1.BackColor = Color.Red;
    }

    if (label8.Text == textBox2.Text)
    {
        textBox2.BackColor = Color.Green;
    }
    else
    {
        textBox2.BackColor = Color.Red;
    }

    if(label9.Text == textBox3.Text)
    {
        textBox3.BackColor = Color.Green;
    }
    else
    {
        textBox3.BackColor = Color.Red;
    }

    if(label11.Text == textBox4.Text)
    {
        textBox4.BackColor = Color.Green;
    }
    else
    {
        textBox4.BackColor = Color.Red;
    }
}

```

61. Captcha Oluşturma



```
private void button4_Click(object sender, EventArgs e)
{
    string[] sembol1 = { "a", "b", "c", "d", "e", "f", "g", "h" };
    string[] sembol2 = { "+", "-", "*", "/", "#", "$", "%" };
    Random captcha = new Random();
    int s1, s2, s3;
    s1 = captcha.Next(0, sembol1.Length);
    s2 = captcha.Next(0, sembol2.Length);
    s3 = captcha.Next(0, 10);
    label13.Text = sembol1[s1].ToString() + sembol2[s2].ToString() + s3.ToString();
}
```

62. Formlar Arası Geçiş

5.

Tıklandığında
form2'ye gider



```
private void button5_Click(object sender, EventArgs e)
{
    Form2 fr = new Form2();
    fr.Show();
    this.Hide();
}
```

63. Formlar Arası Veri Taşıma

6.

Merhaba!

Tıklandığında
textbox'daki veriyi
form2'deki label'a
yazar



6. Merhaba

```
private void button6_Click(object sender, EventArgs e)
{
    Form2 bilgi = new Form2();
    bilgi.mesaj = textBox5.Text;
    bilgi.Show();
    this.Hide();
}
```

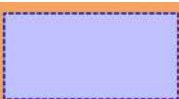
```
public string mesaj;
```

```
private void Form2_Load(object sender, EventArgs e)
{
    label2.Text = mesaj;
}
```

64. Point Yapısı

- Point, x ve y koordinatlarını oluşturabilmek için gerekli alan için kullanılır.

7.



Tıklandığında
picturebox kodda
yazılan lokasyon
yani x ve y
değerlerine
hareket eder.



7.

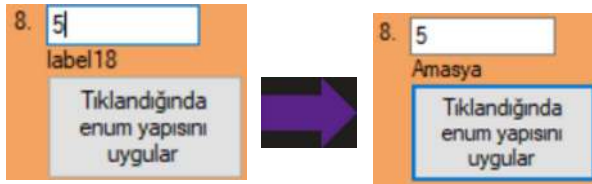
Tıklandığında
picturebox kodda
yazılan lokasyon
yani x ve y
değerlerine
hareket eder.




```
private void button7_Click(object sender, EventArgs e)
{
    Point konum = new Point();
    konum.X = 250;
    konum.Y = 350;
    pictureBox1.Location = konum;
}
```

65. Enum Yapısı

- Enum yapısı, karar yapıların çok fazla olduğu ve verilerin ardışık olarak geldiği alanlarda kullanılır.
- s= (sehirler)plaka; anlamı s, şehirlerin plakadaki sayının değeri şeklindedir.
- Diziler ilk olarak 0 ile başladığından ilk olarak herhangi bir şey atadık.

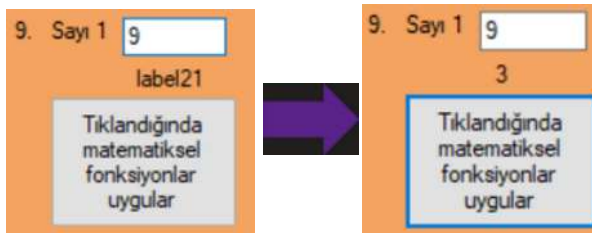


```
enum şehirler { a, Adana, Adıyaman, Afyon, Ağrı, Amasya, Ankara, Antalya, Artvin};
```

```
private void button8_Click(object sender, EventArgs e)
{
    int plaka = Convert.ToInt16(textBox6.Text);
    şehirler s;
    s = (şehirler)plaka;
    label18.Text = s.ToString();
}
```

66. Matematik Fonksiyonları

- Matematik fonksiyonlardan abs; mutlak değerini alma, ceiling; ondalıklı sayıları üste yuvarlama, floor; ondalıklı sayıyı alta yuvarlama, pow; üs alma, sqrt; karekök alma olarak kullanılır.



```
private void button9_Click(object sender, EventArgs e)
{
    double sayi = Convert.ToDouble(textBox7.Text);
    //label21.Text = Math.Abs(sayi).ToString();
    //label21.Text = Math.Ceiling(sayi).ToString();
    //label21.Text = Math.Floor(sayi).ToString();
    //label21.Text = Math.Pow(sayi,4).ToString();
    label21.Text = Math.Sqrt(sayi).ToString();
}
```

67. Dinamik Araçlar

- Autosize, metnin ölçeğine göre boyutlandırma işlemi yapar.

10.

Tıklandığında
dinamik araç
oluşturur



Merhaba

1

2

3

Tıkla

4

```
private void button10_Click(object sender, EventArgs e)
{
    Button btn = new Button();
    Point btnkonum = new Point(400,350);
    btn.Location = btnkonum;
    btn.Name = "Buton1";
    btn.Text = "Tıkla";
    btn.BackColor = Color.Aqua;
    btn.Height = 100;
    btn.Width = 200;
    this.Controls.Add(btn);

    Label lbl = new Label();
    Point lblkonum = new Point(400,100);
    lbl.Location = lblkonum;
    lbl.Name = "Metin1";
    lbl.Text = "Merhaba";
    lbl.BackColor = Color.Yellow;
    lbl.Height = 50;
    lbl.Width = 75;
    this.Controls.Add(lbl);
    lbl.AutoSize = true;

    for (int i = 1; i < 5; i++)
    {
        TextBox txt = new TextBox();
        Point txtkonum = new Point(600, i * 110);
        txt.Location = txtkonum;
        txt.Name = "txt1"+i;
        txt.Text = i.ToString();
        this.Controls.Add(txt);
    }
}
```

10 Dosya İşlemleri

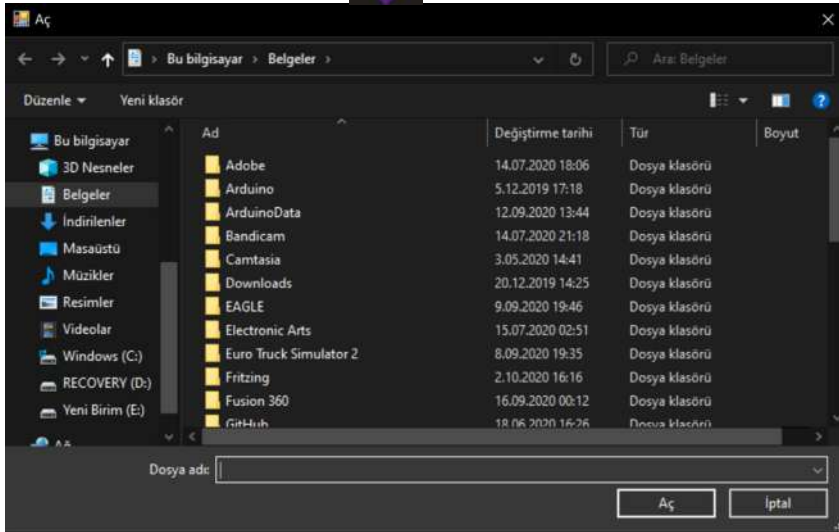
12 Ekim 2020 Pazartesi 22:48

10 Dosya İşlemleri

93. Araçlar - OpenFileDialog / SaveFileDialog / FolderBrowserDialog

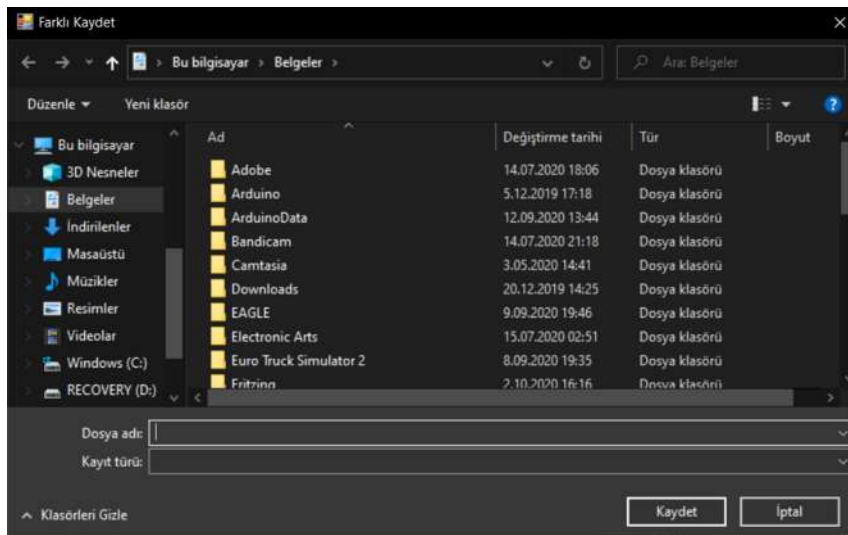
- OpenFileDialog; dosyayı açmak için pencereyi, SaveFileDialog; dosyayı kaydetmek için pencereyi ve SaveBrowserDialog; klasörlerin olduğu pencereyi açar. Üç araçta formun arkasında çalışır.

OpenFileDialog



```
private void button1_Click(object sender, EventArgs e)
{
    openFileDialog1.ShowDialog();
}
```

SaveFileDialog

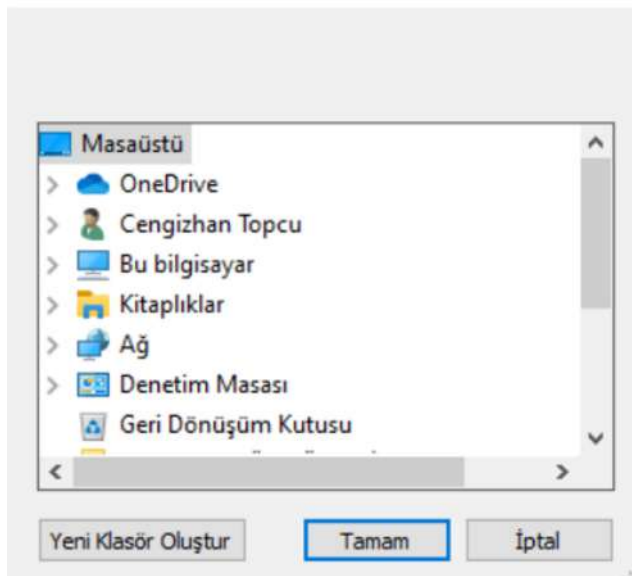


```
private void button2_Click(object sender, EventArgs e)
{
    saveFileDialog1.ShowDialog();
}
```

FolderBrowserDialog



Klasöre Gözet



```
private void button3_Click(object sender, EventArgs e)
{
    folderBrowserDialog1.ShowDialog();
}
```

94. Metin Belgesi Oluşturma

- StreamWriter, dosyaya yazı yazma işlemlerinde kullanılır.

2. Dosya Yolu:	<input type="text"/>	Yol Seç
Dosya Adı:	<input type="text"/>	Oluştur



2. Dosya Yolu: Yol Seç

Dosya Adı: Oluştur

Klasöre Gözet

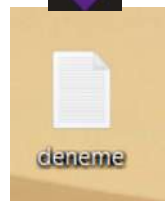
- Masaüstü
- > OneDrive
- > Cengizhan Topcu
- > Bu bilgisayar
- > Kitaplıklar
- > Ağ
- > Denetim Masası
- > Geri Dönüşüm Kutusu

Yeni Klasör Oluştur Tamam İptal



2. Dosya Yolu: Yol Seç

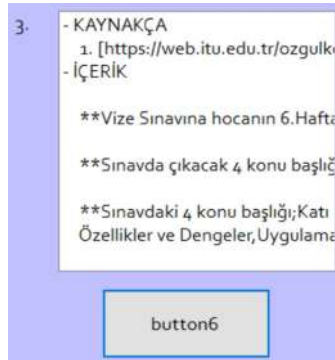
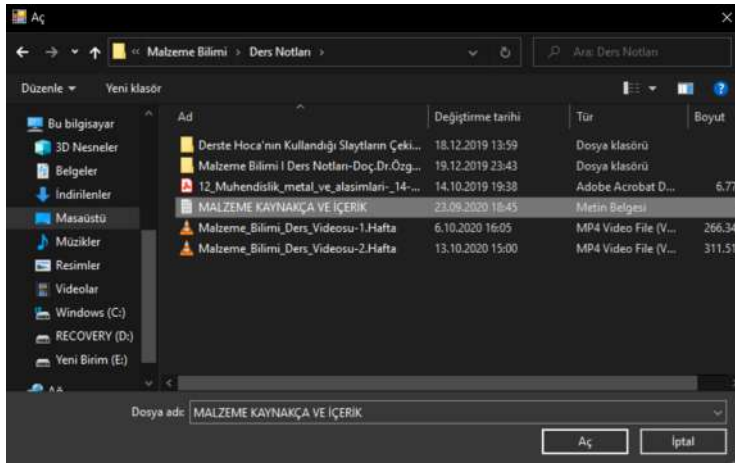
Dosya Adı: Oluştur



```
using System.IO;
string dosyayolu, dosyaadi;
StreamWriter sw;
private void button5_Click(object sender, EventArgs e)
{
    dosyaadi = textBox2.Text;
    sw = File.CreateText(dosyayolu + "\\ " + dosyaadi + ".txt");
    sw.Close();
}

private void button4_Click(object sender, EventArgs e)
{
    if (folderBrowserDialog2.ShowDialog()==DialogResult.OK)
    {
        dosyayolu = folderBrowserDialog2.SelectedPath.ToString();
        textBox1.Text = dosyayolu;
    }
}
```


95. Metin Belgesi Okuma

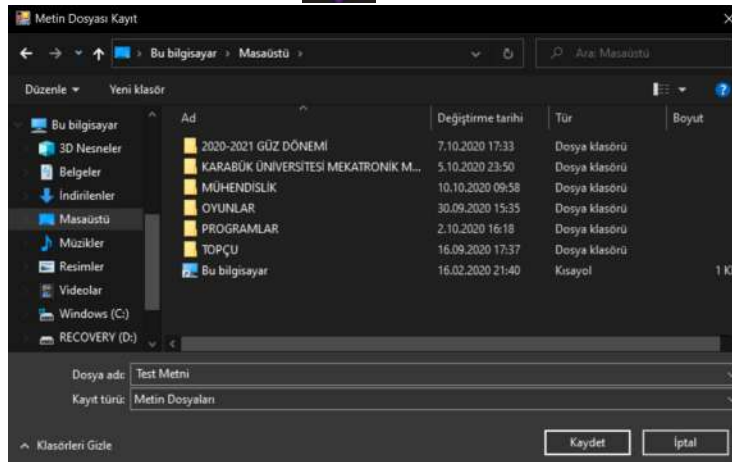


```
using System.IO;
private void button6_Click(object sender, EventArgs e)
{
    if (openFileDialog2.ShowDialog()==DialogResult.OK)
    {
        StreamReader sr = new StreamReader(openFileDialog2.FileName);
        string satir = sr.ReadLine();
        while (satir !=null)
        {
            listBox1.Items.Add(satir);
            satir = sr.ReadLine();
        }
    }
}
```

96. Metin Belgesine Veri Kaydetme

4. Merhabalar benim isimim Alex De Souza.

button7



Kayıt oluşturuldu

Tamam

Test Metni

Test Metni - Not Defteri

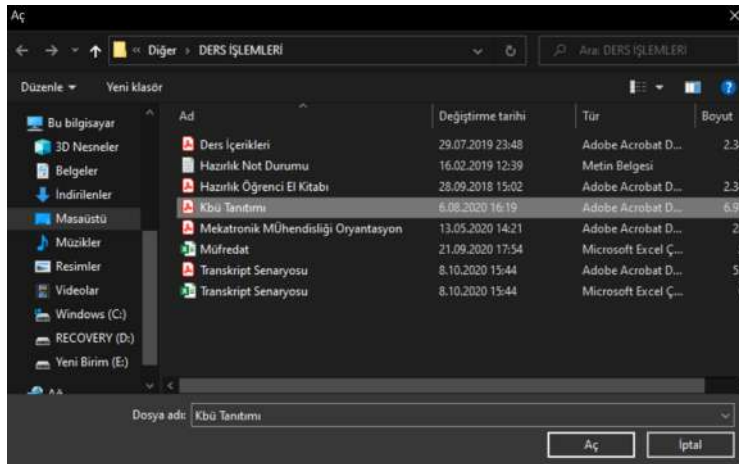
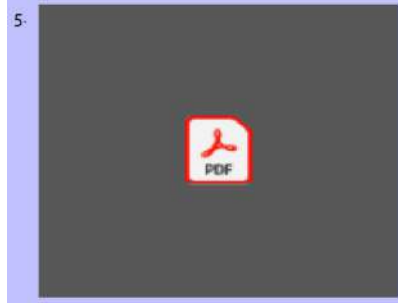
Dosya Düzen Biçim Görünüm Yardım

Merhabalar benim isimim Alex De Souza.

```
using System.IO;
```

```
private void button7_Click(object sender, EventArgs e)
{
    saveFileDialog2.Filter = "Metin Dosyaları|*.txt";
    saveFileDialog2.Title = "Metin Dosyası Kayıt";
    saveFileDialog2.ShowDialog();
    StreamWriter sw = new StreamWriter(saveFileDialog2.FileName);
    sw.WriteLine(richTextBox1.Text);
    sw.Close();
    MessageBox.Show("Kayıt oluşturuldu");
}
```

97. PDF Okuma - Axacropdf Aracı



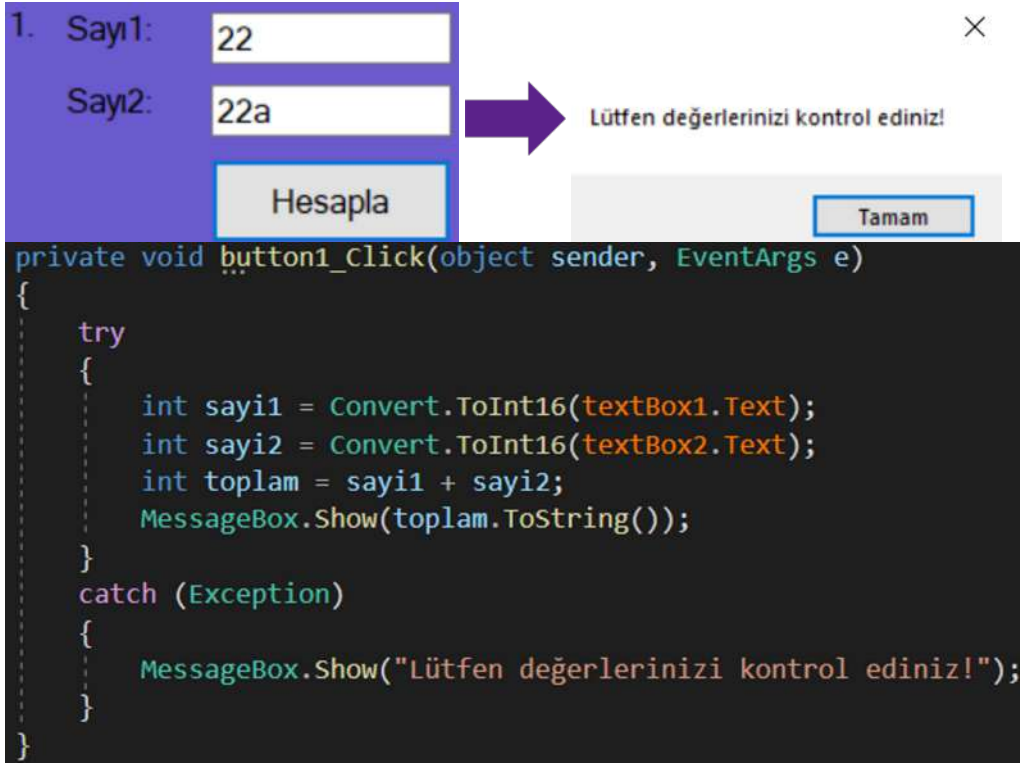
11 Hata Kontrolleri

19 Ekim 2020 Pazartesi 23:54

11 Hata Kontrolleri

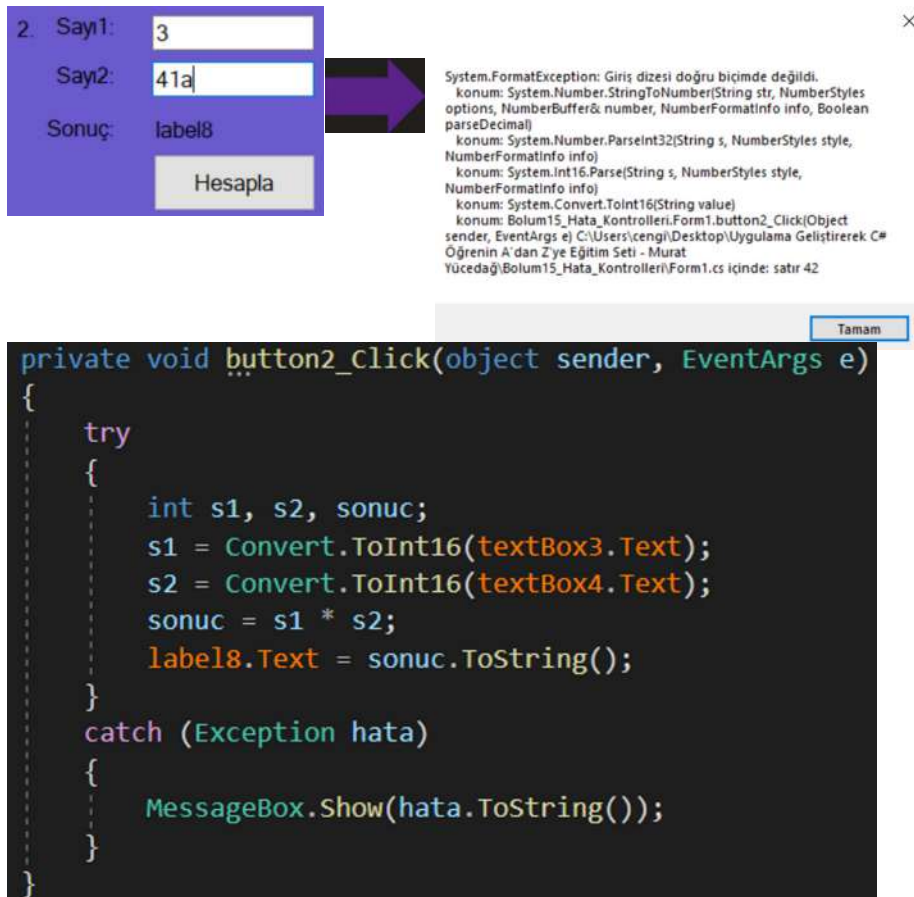
128. Try Catch Bloğu Kullanımı

- Try ile yapmak istediğimizi giriyoruz ve catch kısmında ise hata var ise bu kısım çalışır.



```
private void button1_Click(object sender, EventArgs e)
{
    try
    {
        int sayi1 = Convert.ToInt16(textBox1.Text);
        int sayi2 = Convert.ToInt16(textBox2.Text);
        int toplam = sayi1 + sayi2;
        MessageBox.Show(toplam.ToString());
    }
    catch (Exception)
    {
        MessageBox.Show("Lütfen değerlerinizi kontrol ediniz!");
    }
}
```

129. Exception ve Mesajlar



```
private void button2_Click(object sender, EventArgs e)
{
    try
    {
        int s1, s2, sonuc;
        s1 = Convert.ToInt16(textBox3.Text);
        s2 = Convert.ToInt16(textBox4.Text);
        sonuc = s1 * s2;
        label8.Text = sonuc.ToString();
    }
    catch (Exception hata)
    {
        MessageBox.Show(hata.ToString());
    }
}
```

130. Finally

- Finally, hata olsada olmasada mutlaka çalışan bloktur.



```
private void button3_Click(object sender, EventArgs e)
{
    try
    {
        int s1, s2, sonuc;
        s1 = Convert.ToInt16(textBox5.Text);
        s2 = Convert.ToInt16(textBox6.Text);
        sonuc = s1 * s2;
        label9.Text = sonuc.ToString();
    }
    catch (Exception )
    {
        MessageBox.Show("Hata var burası çalıştı");
    }
    finally
    {
        MessageBox.Show("Finally kodu çalıştı");
    }
}
```




.@cengizhantopcu53

<https://linktr.ee/cengizhantopcu53>