

Sınıflar-Metotlar

Aynı şekilde; program geliştirirken bütün metotları tek bir sınıfın içinde tanımlamak yerine benzer görevleri yerine getiren metotları bir sınıf içerisinde tanımlamak kod tekrarını azaltır, okunabilirliği artırır ve programın daha kolay geliştirilebilmesini sağlar.

Erişim Belirleyicisi: Metot ve özelliklere olan erişimin sınırlarını belirtmektedir.

1. **Private:** Sadece tanımlandığı sınıf içerisinde erişilebilir.
2. **Public:** Her yerden erişilebilir.
3. **Internal:** Sadece bulunduğu projede erişilebilir.
4. **Protected:** Sadece tanımlandığı sınıfta ya da o sınıfı miras alan sınıflardan erişilebilir.

Klavyeden girilen bilgilerler yamuğun alanını hesaplayan yamuk sınıfı üretip, fonksiyon kullanıp hesaplama yapan kod satırlarını yazınız.

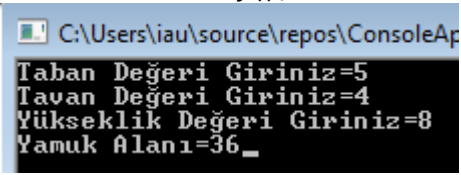
```
namespace ConsoleApp1
{
    class yamuk
    {
        public double tavan;
        public double taban;
        public double yukseklik;

        public void alan()
        {
            Console.WriteLine("Yamuk Alanı=" + (taban+tavan)/2*yukseklik);
        }
    }

    class Program
    {
        static void Main(string[] args)
        {
            yamuk ym2 = new yamuk(); Yamuk Sınıfından değişkenlere erişmek için nesne
            türettik

            Console.WriteLine("Taban Değeri Giriniz=");
            ym2.taban = Convert.ToDouble(Console.ReadLine()); //Taban Bilgisi erişimi
            Console.WriteLine("Tavan Değeri Giriniz=");
            ym2.tavan = Convert.ToDouble(Console.ReadLine()); //Tavan bilgisi erişimi
            Console.WriteLine("Yükseklik Değeri Giriniz=");
            ym2.yukseklik = Convert.ToDouble(Console.ReadLine()); //Yükseklik erişimi
            ym2.alan(); //Alan hesabını yapıp ekrana yazdırma fonksiyonunu çağırma

            Console.ReadKey();
        }
    }
}
```



NOT= Fonksiyon içerisinde void değer döndürmeyip sadece sonucu Main fonksiyonuna yönlendirir. Return döndürme metodunu da tanımlamak için void kullanılmaz tip kullanılır.

```
class yamuk
{
    public double tavan;
```

```

        public double taban;
        public double yukseklik;

    public double alan()
    {
        return (taban+tavan)/2*yukseklik;
    }
}

class Program
{
    static void Main(string[] args)
    {
        yamuk ym2 = new yamuk(); //Yamuk Sınıfından değişkenlere erişmek için nesne
        Console.Write("Taban Değeri Giriniz=");
        ym2.taban = Convert.ToDouble(Console.ReadLine()); //Taban Bilgisi erişimi
        Console.Write("Tavan Değeri Giriniz=");
        ym2.tavan = Convert.ToDouble(Console.ReadLine()); //Tavan bilgisi erişimi
        Console.Write("Yükseklik Değeri Giriniz=");
        ym2.yukseklik = Convert.ToDouble(Console.ReadLine()); //Yükseklik erişimi
        Console.Write( ym2.alan());
    }
}

```

Dairenin çevresini ve alanını hesaplayan programı metod kullanarak yazınız.

```

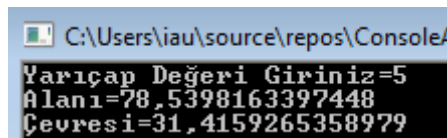
class daire
{
    public double yarıcap;

    public double cevre()
    {
        return (2 * Math.PI * yarıcap);
    }

    public double alan()
    {
        return (Math.PI * Math.Pow(yarıcap, 2));
    }
}

class Program
{
    static void Main(string[] args)
    {
        daire daire = new daire();
        Console.Write("Yarıçap Değeri Giriniz=");
        daire.yarıcap = Convert.ToDouble(Console.ReadLine());
        Console.WriteLine("Alanı=" + daire.alan());
        Console.WriteLine("Çevresi=" + daire.cevre());
    }
}

```



```

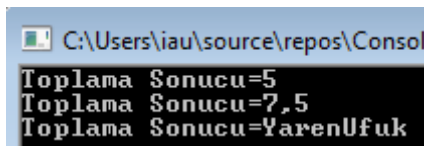
C:\Users\iau\source\repos\ConsoleApp\ConsoleApp
Yarıçap Değeri Giriniz=5
Alanı=78.5398163397448
Çevresi=31.4159265358979

```

Overloading (Metodların aşırı yüklenmesi)

```
class Matematik
{
    public void toplama(int a,int b)
    {
        Console.WriteLine("Toplama Sonucu=" + (a + b));
    }
    public void toplama(double a, double b)
    {
        Console.WriteLine("Toplama Sonucu=" + (a + b));
    }
    public void toplama(string a, string b)
    {
        Console.WriteLine("Toplama Sonucu=" + (a + b));
    }
}

class Program
{
    static void Main(string[] args)
    {
        Matematik m1 = new Matematik();
        m1.toplama(2, 3);
        m1.toplama(2.3, 5.2);
        m1.toplama("Yaren", "Ufuk");
    }
}
```



Sınıf kullanarak hesap makinesi programı kod satırlarını yazınız.

```
class Matematik
{
    public double sayi1, sayi2;

    public void toplama()
    {
        Console.WriteLine("Toplama Sonucu=" + (sayi1 + sayi2));
    }
    public void Cikarma()
    {
        Console.WriteLine("Çıkarma Sonucu=" + (sayi1 - sayi2));
    }
    public void Carpma()
    {
        Console.WriteLine("Çarpma Sonucu=" + (sayi1*sayi2));
    }
    public void Bolme()
    {
        Console.WriteLine("Bölme Sonucu=" + (sayi1 / sayi2));
    }
}
```

```

class Program
{
    static void Main(string[] args)
    {
        int islem;
        Matematik matematik = new Matematik();
        do
        {
            Console.Write("1.Sayıyı Giriniz=");
            matematik.sayi1 = Convert.ToDouble(Console.ReadLine());
            Console.Write("2.Sayıyı Giriniz=");
            matematik.sayi2 = Convert.ToDouble(Console.ReadLine());
            Console.WriteLine("1.Toplama 2.Çıkarma 3.Çarpma 4.Bölme .....0
ile çıkınız");
            islem = Convert.ToInt16(Console.ReadLine());
            switch (islem)
            {
                case 1: matematik.toplama(); break;
                case 2: matematik.Cikarma(); break;
                case 3: matematik.Carpma(); break;
                case 4: matematik.Bolme(); break;
            }
        } while (islem != 0);
    }
}

```

İdeal kiloyu hesaplayan programın kod satırlarını yazınız.

```

class İdeal
{
    public double boy, yıl,kilo,k,yas,ideal;
    public string cinsiyet;

    public void hesapla()
    {
        yas = DateTime.Now.Year - yıl;
        ideal= ((boy / 100 + yas) / 10) *k;

        if (cinsiyet == "e" || cinsiyet == "E")
        {
            k = 0.9;
            ideal = (boy - 100 + yas / 10) * k;
            Console.WriteLine("İdeal Kilonuz=" + ideal);
        }
        else
        {
            k = 0.8;
            ideal = (boy - 100 + yas / 10) * k;
            Console.WriteLine("İdeal Kilonuz" + ideal);
        }
    }
}

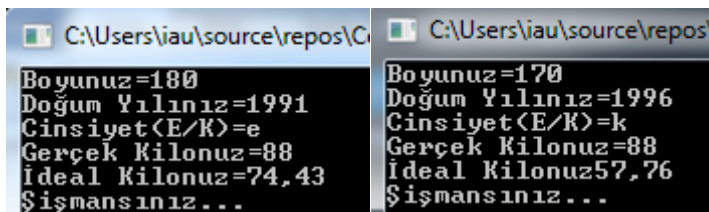
```

```

class Program
{
    static void Main(string[] args)
    {
        İdeal ideal = new İdeal();
        Console.Write("Boyunuz=");
        ideal.boy = Convert.ToDouble(Console.ReadLine());
        Console.Write("Doğum Yılıınız=");
        ideal.yil = Convert.ToDouble(Console.ReadLine());
        Console.Write("Cinsiyet(E/K)=");
        ideal.cinsiyet = Console.ReadLine();
        Console.Write("Gerçek Kilonuz=");
        ideal.kilo = Convert.ToDouble(Console.ReadLine());

        if(ideal.ideal == ideal.kilo)
        {
            ideal.hesapla();
            Console.WriteLine("Bravo...İdeal Kilodasınız");
        }
        else if(ideal.ideal > ideal.kilo)
        {
            ideal.hesapla();
            Console.WriteLine("Zayıfsınız...");
        }
        else if(ideal.ideal < ideal.kilo)
        {
            ideal.hesapla();
            Console.WriteLine("Şişmansınız...");
        }
    }
}

```



Sayı tut metodu kullanarak klavyeden 1 gönderilince tek sayı 2 gönderilince çift sayı üreten kod satırlarını yazınız.

```

class sayitut
{
    public int sayi1, sayi2, gelen;

    public void SayiTut()
    {
        Random rastgele = new Random();
        if (gelen == 1) {
            do {
                sayi1 = rastgele.Next(1, 101);
                sayi2 = sayi1 % 2;
            } while (sayi2 != 1);
            Console.WriteLine("Tek sayı=" + sayi1);
        }
        if (gelen == 2) {
            do {
                sayi1 = rastgele.Next(1, 101);
                sayi2 = sayi1 % 2;
            } while (sayi2 != 0);
            Console.WriteLine("Çift sayı=" + sayi1);
        }
    }
}

```

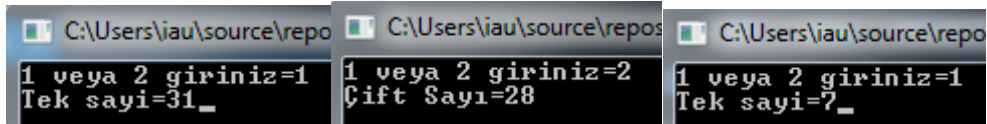
```

        } while (sayi2 != 0);
        Console.Write("Çift Sayı=" + sayi1);}
    }

    }

class Program
{
    static void Main(string[] args)
    {
        sayitut tut = new sayitut();
        Console.Write("1 veya 2 giriniz=");
        tut.gelen = Convert.ToInt16(Console.ReadLine());
        tut.SayiTut();
    }
}

```



Bir araca ait gittiği yol ve gidiş süresi klavyeden girildiğinde ortalama hızını hesaplayıp ekrana yazan C# konsol uygulamasını yazınız.

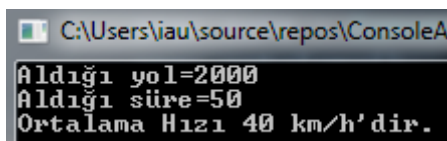
```

class Arac
{
    double yol;
    double sure;
    double ortalama()
    {
        return yol / sure;
    }
}

class Program
{
    static void Main(string[] args)
    {
        Arac arac = new Arac();
        Console.Write("Aldığı yol=");
        arac.yol = Convert.ToDouble(Console.ReadLine());
        Console.Write("Aldığı süre=");
        arac.sure = Convert.ToDouble(Console.ReadLine());
        arac.ortalama();
        Console.WriteLine("Ortalama Hızı {0} km/h'dir." , arac.ortalama());

        Console.ReadKey();
    }
}

```



Static metot tanımlamaları ve parametre kullanımı

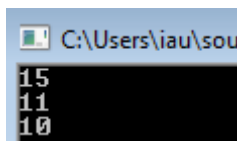
namespace ConsoleApp1

```
{
    class sinif1
    {
        public int topla1(int sayi1,int sayi2)
        {
            return sayi1 + sayi2;
        }
    }
    class sinif2
    {
        public static int topla2 (int sayi1,int sayi2)
        {
            return sayi1 + sayi2;
        }
    }
    class Program
    {
        public static int topla(int sayi1,int sayi2)
        {
            return sayi1 + sayi2;
        }
        static void Main(string[] args)
        {
            Console.WriteLine(topla(5, 10));

            //*****SINIF1 DEN ÇAĞIRMA*****//
            sinif1 snf1 = new sinif1();

            Console.WriteLine(snf1.topla1(4,7));

            //*****SINIF2 DEN ÇAĞIRMA*****//
            Console.WriteLine(sinif2.topla2(6, 4));//Static tanımlama yapıldığı zaman
            yeni bir nesne oluşturmaya ihtiyaç yoktur.
        }
    }
}
```



Bir otoparkta park eden taksinin 1 saati 5TL,minibüsün 6TL,ticarinin 6.5 TL'dir. Araçların 1 saatten fazla kaldıklarından ödeyeceği fark sırayla %20,%21,5,%25 tir. Klavyeden girilen araç tipi ve saat bilgileriyle ödeyeceği ücret miktarını hesaplayan kod satırlarını yazınız.

```
namespace otoparkHesap
{
    2 references
    class tutar
    {
        public int saat;

        1 reference
        public double otomobilHesap()
        {
            double ucretToplam = 5;

            for (int i = 1; i < saat; i++)
            {
                ucretToplam = ucretToplam * 1.20;
            }

            return ucretToplam;
        }

        1 reference
        public double minibusHesap()
        {
            double ucretToplam = 6;

            for (int i = 1; i < saat; i++)
            {
                ucretToplam = ucretToplam * 1.215;
            }

            return ucretToplam;
        }
    }
}
```

```

1 reference
public double ticariHesap()
{
    double ucretToplam = 6.5;

    for (int i = 1; i < saat; i++)
    {
        ucretToplam = ucretToplam * 1.25;
    }

    return ucretToplam;
}

0 references
class Program
{
    0 references
    static void Main(string[] args)
    {
        for(;;)
        {

            tutar tr = new tutar();
            Console.WriteLine("Araç Tipini giriniz. \n 1-Otomobil \n 2-Minibüs \n 3-Ticari");
            int tip = Convert.ToInt32(Console.ReadLine());

            if (tip > 3 || tip < 1)
                break;

            Console.WriteLine("süre giriniz");
            tr.saat = Convert.ToInt32(Console.ReadLine());

            switch (tip)
            {
                case 1: Console.WriteLine("Ödenecek Tutar = " + tr.otomobilHesap()); break;
                case 2: Console.WriteLine("Ödenecek Tutar = " + tr.minibusHesap()); break;
                case 3: Console.WriteLine("Ödenecek Tutar = " + tr.ticariHesap()); break;

                default:
                    break;
            }

            Console.ReadKey();
        }
    }
}

```

Windows Form Uygulamaları

Form uygulamalarında en bilinmesi gereken en önemli metot form geçişlerini sağlayan **Form.Show()** metodudur. Form içerisine yazılan **this.Hide()** metodu ise diğer forma geçildiği zaman eski formu

gizlemektedir. **this.Close()** metodu ise o formu direk olarak kapatan metottur. Ancak bu metodu ana formda uygularsak uygulamayı direk olarak kapatır.

Önemli özelliklerden bir tanesi ise kod ile label eklenmesidir.

```
Label label2 = new Label();
this.Controls.Add(label2);
label2.Text = "Selam...";
```

AutoSize() özelliği ise label içeri otomatik boyutlandırmaktadır. True ve false olarak iki değer almaktadır. True dersek otomatik boyutlandırmaktadır.

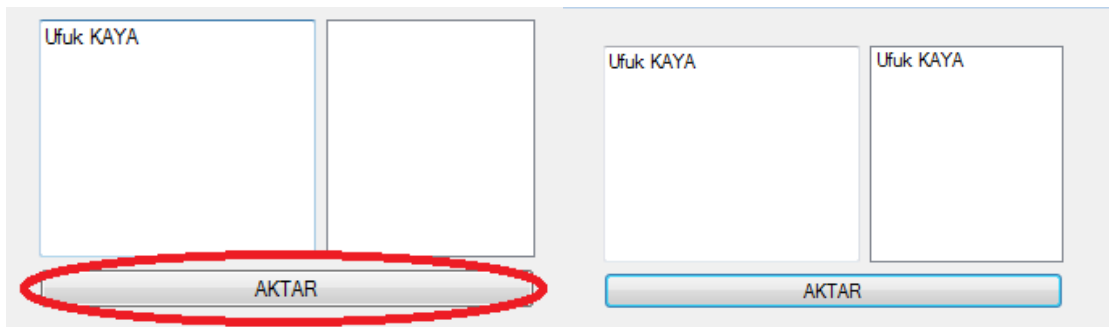
Font() sınıfından da label dizayn edebiliriz. **ForeColor()** özelliği kullanarak değişik renkleri kullanabiliriz.

```
label2.AutoSize = true;
label2.Font = new Font("Comic Sans Ms", 15, FontStyle.Italic);
label2.ForeColor = Color.Red;
```

TextBox içerisindeki verileri Listbox içerisine yazan kod satırlarını yazınız.

```
public partial class Form1 : Form
{
    public Form1()
    {
        InitializeComponent();
    }

    private void btnAktar_Click(object sender, EventArgs e)
    {
        string metin = textBox1.Text;
        listBox1.Items.Add(metin);
    }
}
```



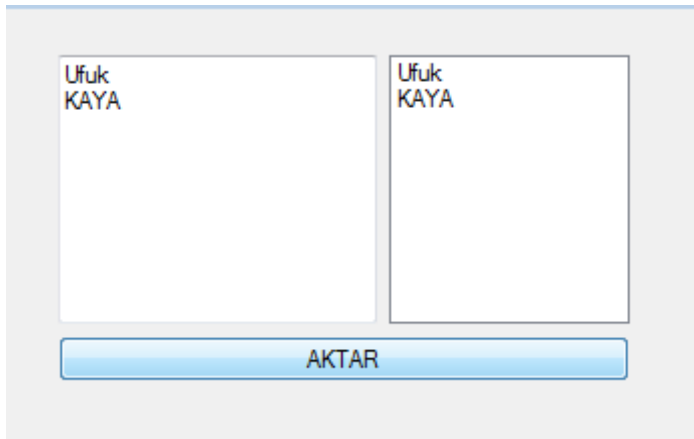
Farklı olarak kullanacağımız metotta aşağıdaki kodda mevcuttur. Bu kodla da alt alta yazabilmemiz gerektiğinde kullanırız.

```
private void btnAktar_Click(object sender, EventArgs e)
{
    for(int i = 0; i < textBox1.Lines.Length; i++)
    {
```

```

        listBox1.Items.Add(textBox1.Lines[i]);
    }
}

```

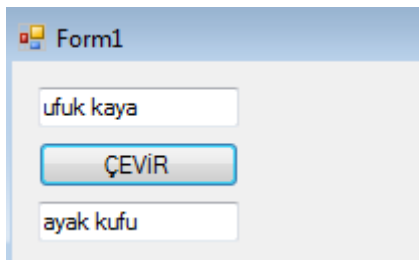


Textbox içerisindeki metni tersten yazan kod satırlarını yazınız.

```

private void btnCevir_Click(object sender, EventArgs e)
{
    char[] str=textBox1.Text.ToCharArray();//Metni karakter karakter alıp
    diziye atma islemi
    Array.Reverse(str);//Diziyi tersine çevirme metodu
    string strTers = new string(str); //TextBox içerisine yalnızca string
    ifadeler yazıldığı için stringe çevirme işlemi yaptık.
    textBox2.Text = strTers; //TextBox içerisine yazdırma
}
}

```



Basit bir hesap makinesi yapan programın kod satırlarını yazınız.

```

private void btnTopla_Click(object sender, EventArgs e)
{
    int sayi1 = Convert.ToInt16(tbSayi1.Text);
    int sayi2 = Convert.ToInt16(tbSayi2.Text);
    int sonuc;
    sonuc = sayi1 + sayi2;
    tbSonuc.Text = sonuc.ToString();
}

```

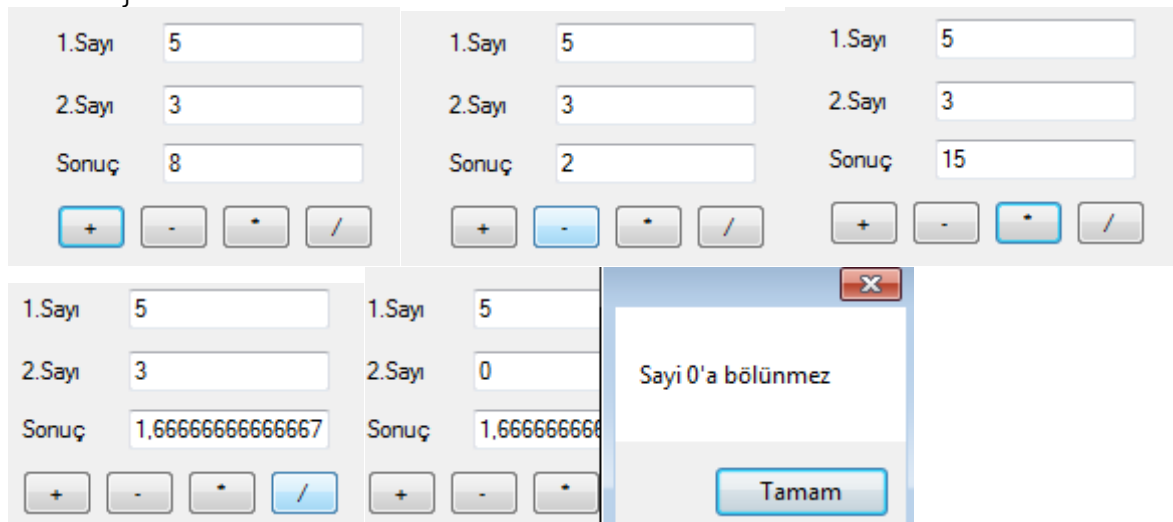
```

private void btnCikar_Click(object sender, EventArgs e)
{
    int sayi1 = Convert.ToInt16(tbSayi1.Text);
    int sayi2 = Convert.ToInt16(tbSayi2.Text);
    int sonuc;
    sonuc = sayi1 - sayi2;
    tbSonuc.Text = sonuc.ToString();
}

private void btnCarp_Click(object sender, EventArgs e)
{
    int sayi1 = Convert.ToInt16(tbSayi1.Text);
    int sayi2 = Convert.ToInt16(tbSayi2.Text);
    int sonuc;
    sonuc = sayi1 * sayi2;
    tbSonuc.Text = sonuc.ToString();
}

private void btnBol_Click(object sender, EventArgs e)
{
    int sayi1 = Convert.ToInt16(tbSayi1.Text);
    int sayi2 = Convert.ToInt16(tbSayi2.Text);
    double sonuc;
    if (sayi2 == 0) { MessageBox.Show("Sayi 0'a bölünmez"); }
    else
    {
        sonuc = sayi1 / sayi2;
        tbSonuc.Text = sonuc.ToString();
    }
}

```



İdeal kilo bulma uygulaması yapan uygulamanın kod satırlarını yazınız.

```

private void btnHesap_Click(object sender, EventArgs e)
{
    double kilo, boy, yas, ideal;

    kilo = Convert.ToDouble(tbKilo.Text);
    boy = Convert.ToDouble(tbBoy.Text);
    yas = DateTime.Now.Year - Convert.ToDouble(tbDogum.Text);

    if (rbErkek.Checked)
    {
        ideal = (boy - 100 + yas / 10) * 0.9;
    }
}

```

```

        tbIdeal.Text = ideal.ToString();

        if (ideal == kilo){ MessageBox.Show("Tebrikler");}
        else if (ideal > kilo) { MessageBox.Show("Zayıfsınız..."); }
        else if (ideal < kilo) { MessageBox.Show("Şişmansınız..."); } }

        if (rbKadin.Checked)
        {
            ideal = (boy - 100 + yas / 10) * 0.8;
            tbIdeal.Text = ideal.ToString();

            if (ideal == kilo) { MessageBox.Show("Tebrikler"); }
            else if (ideal > kilo) { MessageBox.Show("Zayıfsınız..."); }
            else if (ideal < kilo) { MessageBox.Show("Şişmansınız..."); }
        }
    }
}

```

Radio Button Kullanımı

RadioButton.Checked metodu radio seçildikten sonraki tetiklemeleri gösterir.Yani seçildikten sonra button a basıldıktan sonra aktif olur.

RadioButton.CheckedChanged ise radio button seçildiği anda tetiklenir.Yani herhangi bir button kullanılmaya gerek kalmadan aktif olur.

```

private void radioButton1_CheckedChanged(object sender, EventArgs e)
{
    if (radioButton1.Checked) { label1.Text = radioButton1.Text; }
    if (radioButton2.Checked) { label1.Text = radioButton2.Text; }
    if (radioButton3.Checked) { label1.Text = radioButton3.Text; }
}

```

```

private void btnGoster_Click(object sender, EventArgs e)
{
    if (radioButton1.Checked) { label1.Text = radioButton1.Text; }
    if (radioButton2.Checked) { label1.Text = radioButton2.Text; }
    if (radioButton3.Checked) { label1.Text = radioButton3.Text; }
}

```

Form ekranında 10-20-30-40 butonları ve artır-azalt radio butonları olacak.Sayı butonlarına basıldığında textboxtaki sayısı tıklama kadar artıran kod satırlarını yazınız.

```

public partial class Form1 : Form
{
    int sayi = 0;
    public Form1()
    {
        InitializeComponent();
    }

    private void btn10_Click(object sender, EventArgs e)
    {
        if (rbArtir.Checked)
        {
            sayi = sayi + 10;
            tbSayi.Text = sayi.ToString();
        }

        if (rbAzalt.Checked)
        {
            sayi = sayi - 10;
            tbSayi.Text = sayi .ToString();
        }
    }

    private void btn20_Click(object sender, EventArgs e)
    {
        if (rbArtir.Checked)
        {
            sayi = sayi + 20;
            tbSayi.Text = sayi.ToString();
        }

        if (rbAzalt.Checked)
        {
            sayi = sayi - 20;
            tbSayi.Text = sayi.ToString();
        }
    }

    private void btn30_Click(object sender, EventArgs e)
    {
        if (rbArtir.Checked)
        {
            sayi = sayi + 30;
            tbSayi.Text = sayi.ToString();
        }

        if (rbAzalt.Checked)
        {
            sayi = sayi - 30;

```

```

        tbSayi.Text = sayi.ToString();
    }
}

private void btn40_Click(object sender, EventArgs e)
{
    if (rbArtir.Checked)
    {
        sayi = sayi + 40;
        tbSayi.Text = sayi.ToString();
    }

    if (rbAzalt.Checked)
    {
        sayi = sayi - 40;
        tbSayi.Text = sayi.ToString();
    }
}
}

```

Butona her basıldığında butonun içerisindeki sayıyı artıran kod satırını yazınız.

```

public partial class Form1 : Form
{
    int sayi = 0;
    public Form1()
    {
        InitializeComponent();
    }

    private void btnTrigger_Click(object sender, EventArgs e)
    {
        sayi = sayi + 1;
        btnTrigger.Text = sayi.ToString();
        if (sayi == 10)
        {
            MessageBox.Show("Yeeeeettteerrr....");
            Application.Exit();
        }
    }
}

```



CheckBox Kullanımı

checkBox1.Checked() butonla tetiklenen metottur. **checkBox1.CheckedChanged()** ise sadece seçildiğinde tetiklenir. Button tetiklemesine ihtiyaç yoktur.

```
private void btnHesapla_Click(object sender, EventArgs e)
{
    double tutar, toplam;
    tutar = Convert.ToDouble(textBox1.Text);
    if (checkBox1.Checked == true)
    {
        toplam = tutar - (tutar * 0.2);
        label2.Text = toplam.ToString();
    }
    else { label2.Text = tutar.ToString(); }
}
```

Tutar

☒ %20 indirim uygula

1600

Bir restoranda verilen siparişleri fiyatlarına göre hesaplayıp gösteren bir uygulama.

```
private void btnHesapla_Click(object sender, EventArgs e)
{
    double toplam,sandvic,hamburger,kola ;

    if (checkBox1.Checked == true)
    {
        toplam = 10;
        label2.Text = toplam.ToString();
    }
    if (checkBox2.Checked == true)
    {
        toplam = 15;
        label2.Text = toplam.ToString();
    }
}
```

```

    }
    if (checkBox3.Checked == true)
    {
        toplam = 5;
        label2.Text = toplam.ToString();
    }

    if (checkBox1.Checked == true && checkBox2.Checked == true)
    {
        toplam = 10 + 15;
        label2.Text = toplam.ToString();
    }
    if (checkBox1.Checked == true && checkBox3.Checked == true)
    {
        toplam = 10 + 5;
        label2.Text = toplam.ToString();
    }

    if (checkBox2.Checked == true && checkBox3.Checked == true)
    {
        toplam = 15 + 5;
        label2.Text = toplam.ToString();
    }

    if (checkBox2.Checked == true && checkBox3.Checked == true &&
checkBox1.Checked == true)
    {
        toplam = 15 + 5 + 10;
        label2.Text = toplam.ToString();
    }

    if (checkBox2.Checked == false && checkBox3.Checked == false &&
checkBox1.Checked == false)
    {
        MessageBox.Show("Lütfen Seçim Yapınız"); }

```

Rastgele iki sayı üretilip işlem yaptırarak ve bize 10 hak verip sonuç tahmin ettiren programın kod satırlarını yazınız.

```

public partial class Form1 : Form
{
    double sonuc;
    string isaret;
    int hak = 10, tahmin, skor=0;
    public Form1()
    {
        InitializeComponent();
    }

    private void button1_Click(object sender, EventArgs e)
    {
        int sayi1, sayi2, islem;

```

```

Random rastgele = new Random();
tahmin = Convert.ToInt32(tbTahmin.Text);
sayi1 = rastgele.Next(1, 10);
sayi2 = rastgele.Next(1, 10);
islem = rastgele.Next(0, 4);

if (islem == 0) { sonuc = sayi1 + sayi2; isaret = islem.ToString(); isaret
= "+"; }
else if (islem == 1) { sonuc = sayi1 - sayi2; isaret = islem.ToString();
isaret = "-"; }
else if (islem == 2) { sonuc = sayi1 * sayi2; isaret = islem.ToString();
isaret = "*"; }
else if (islem == 3) { sonuc = sayi1 / sayi2; isaret = islem.ToString();
isaret = "/"; }

if (sonuc == tahmin)
{
    skor++;
    label1.Text = "Cevabınız Doğru" + " " + sayi1 + " " + isaret + " " +
sayi2 + " = " + sonuc ; }
else
{
    hak--;
    label1.Text = "Cevabınız Yanlış" + " " + sayi1 + " " + isaret + " " +
sayi2 + " = " + sonuc + " Kalan Hakkınız=" + hak; ;
    if (hak == 0)
    {
        MessageBox.Show("Üzgünüm Kaybettiniz...Skorunuz=" + skor);
    }
}
tbTahmin.Clear();
}

```

Cevabınız Yanlış $8 / 7 = 1$ Kalan Hakkınız=9 TAHMIN ET	Cevabınız Yanlış $7 * 5 = 35$ Kalan Hakkınız=8 TAHMIN ET
Cevabınız Yanlış $5 + 4 = 9$ Kalan Hakkınız=1 TAHMIN ET	Üzgünüm Kaybettiniz...Skorunuz=0 Tamam