

## CONSOLE APP için DELEGATE KULLANIMI

### Return kullanılarak ; Tek bir fonksiyon kullanımı

```
namespace Delegate
{
    class Program
    {
        public delegate int Fonksiyon(int x, int y);
        static int Topla(int x,int y)
        {
            return x + y;
        }
        static int Carp(int a, int b)
        {
            return a * b;
        }

        static void Main(string[] args)
        {
            int sonuc;
            Fonksiyon fnk = new Fonksiyon(Carp);
            sonuc=fnk(22,33);
            Console.WriteLine("Sonuc:" + sonuc);
        }
    }
}
```

### Birden fazla fonksiyonun delegate'e atanıp kullanımı

```
namespace Delegate
{
    class Program
    {
        public delegate void Fonksiyon(int x, int y);
        static void Topla(int x,int y)
        {
            Console.WriteLine(x + y);
        }
        static void Carp(int a, int b)
        {
            Console.WriteLine(a * b);
        }

        static void Main(string[] args)
        {
            int sonuc;
            Fonksiyon fnk = new Fonksiyon(Carp);
            fnk += Topla;
            fnk(22,33);

            Console.WriteLine("Sonuc:");
        }
    }
}
```

**ÇIKTI:**

**726 (ÇARP)**

**55 (TOPLA)**

## FORM İÇİN DELEGATE KULLANIMI

### Bir Formdan Diğetine veri aktarma.(delegate ve event kullanarak)

Öncelikle Visual Studio da bir WindowsForms oluşturalım ve bu Form1 olsun. Formumuza önce bir Listbox ekleyelim sonrada bir button ve textbox. daha sonra **SolutionExplorer>Add>WindowsForms** diyerek projemize yeni bir form ekleyelim.

bu formumuza ise br textbox ve button ekleyelim. Şimdi yapacaklarımızı şöyle basit bir şekilde düşünelim. Form1 den button a basınca yeni form açılsın ve bizim textbox ın içine yazdığımızı açılan formun textbox ına yazsın. Sonra burda değiştirdiğimiz bu textbox ın içeriğini form2 de bulunan button ile form1 deki listbox'a yazalım. böylece form1 den form2 ye veri atmış ve form2 den de form1 e veri atmış oluyoruz.

şimdi Form1 için yazmamız gereken kodlara geçelim.

form1.cs

```
private void button1_Click(object sender, EventArgs e)
{
    //Burda form2 türünden yeni bir nesne oluşturuyoruz.

    Form2 frm = new Form2();
    //Burda Form2 altındaki textbox a erişip bu formdaki textbox ın değerini açacağımız
    formdakine yazıyoruz.
    frm.textBox1.Text= textBox1.Text;
    //burda olayımızı tetikliyoruz. ve olayımızın kullanacağı fonk gösteriyoruz.
    frm.BenimEventim+= new Form2.BenimDelegem(VeriYaz);
    //formumuzu gösteriyoruz. Burda frm.Show da kullanabilirdik . showdialog kullanarak
    //sadece bir form açık ken buttona yeniden basılısada 2.bir form oluşturulmasını
    //engelliyoruz
    frm.ShowDialog();
}
//VeriYaz fonk tanımlanıyor. Fonksiyonumuz textbox tipinde değişken alıyor.
Void VeriYaz(TextBox txtBox)
{
    // form2den gelen textbox değişkeninin değerini listbox a ekliyoruz.
    Listbox1.Items.Add(txtBox.Text);
}
```

form2.cs

```
// BenimDelegem isimli delegemizi tanımlıyoruz. Parametre olarak TextBox tipinde bir
//değişken alacağını söylüyoruz.
public delegate void BenimDelegem(TextBox txtbox);
// BenimDelegem tipinde BenimEventim isimli event oluşturuyoruz.
public event BenimDelegem BenimEventim;

private void button1_Click(object sender, EventArgs e)
{
// buttona tıkladığımızda BenimEventim adlı olayımıza formumuzdaki textBox1 i atıyoruz.

this.BenimEventim(textBox1);
}
```

Şimdi Genel kurgumuzu yukarıda anlatmıştık fakat bu delegate ve event ne işe yarıyor birazda onlardan bahsedelim. burda form2 de önce bir delegate tanımlıyor ve bunun tipinde bir event oluşturuyor. hemen aşağısında ise event olayında kullanacağımız form2 objesini olayımıza ekliyoruz.

form1 de ise

**frm.BenimEventim+= new Form2.BenimDelegem(VeriYaz);**

bu satırla form2 de tanımladığımız event e erişerek bu olayın veriyaz fonksiyonunu kullacağını söylüyor ve olayı tetikliyoruz.

Ayrıca **this.BenimEventim(textBox1);** kodu ile event imize attığımız textBox1 değerini Veriyaz Fonksiyonuna geçirmiş oluyoruz. ve bu değeri içerde kullanıyoruz. Aşağıda Projemizi Çalıştırılmış Halini Görmektesiniz.

