

Konular

- Delegates
- Events
- Windows Forms

Delegates (Delegeler)

- Delege bir işaretçidir.
- Bir delege referansı bir metoda olarak tutan bir referans tip değişkenidir. Referans çalışma zamanında (runtime) değiştirilebilir.
- Yordamlara aracılık eder ve onları temsil eder.
- Delegeleri, çalışma anında planda olmayan bir yordamı yürütmek için veya bir olay gerçekleştiği zaman ilgili yordamları haberdar etmek için kullanırız.
- Tüm delegeler, System.Delegate sınıfından türetilir.

Delegates (Delegeler)

- Değişkenler gibi metodlar da bellek üzerinde özel bir alanda yaşar.
- Değişken pointerleri gibi metodların başlangıç adresini işaret eden pointerler bulunur. Bunlara metod işaretçileri (function pointers) denilir. İstenirse metodlar, hafıza üzerinden sadece isimleriyle değil bu pointerler sayesinde de çağrılabilir.
- .NET, fonksiyon işaretçileriyle aynı amaca sahip tür güvenliği sağlayan delege (delegate) isimli işaretçiler sunar.

Delegates (Delegeler)

- Yazılım dilinde ise metodlara aracılık eden, onları temsil eden, onlara erişmemizi sağlayan çalışma zamanında tanımlanan yapılardır.
- Delegate, daha önceki function pointerler'den farklı olarak sadece statik metodları işaret etmez, hem statik hem de nesne (instance) metodları işaret eder.

Delegates (Delegeler)

- İki yerde ihtiyaç duyabiliriz:
 - Bazen çalışma zamanında planda olmayan bir metodu yürütmek zorunda kalabiliriz. Bunun için delegateler kullanılır.
 - Bir olay(event) gerçekleştiği zaman bir veya daha fazla metodu, gerçekleşen olaydan haberdar etmek için delegeler kullanırız. Bu sayede, delegeler, eventlerle anlam kazanır.

Delege İşlemler

- Bir delege işlemi üç adımdan oluşur.
 1. Tanımlama(Declaration)
 2. Örneklem, gerçekleşme (Instantiation)
 3. Çağırma(Invocation)

Delege tanımlama

- Delege tanımlama, delege tarafından refere edilecek metodu belirtmektir.
- Delege sadece aynı imzaya (yapıya - signature) sahip bir metodu refere edebilir.

```
public delegate int MyDelegate (string s);
```

```
delegate <return type> <delegate-name> <parameter list>
```

delegate anahtarından sonra döndüreceği değer, delege ismi ve delege'ye ait parametre sayısı ve türüne girilir. Bu ifade bir imzadır. (signature)

Delege Örnekleme ve Çağırma

```
class Program
{
    //1.tanımlama
    public delegate int MyDelegate(int x);

    static int KareAl(int sayi)
    {
        return sayi * sayi;
    }

    static void Main(string[] args)
    {
        MyDelegate delege1 = new MyDelegate(KareAl); //2.declaration
        int y = delege1(4); //3.Invocation

        Console.WriteLine(y);

        Console.ReadKey();
    }
}
```

- KareAl fonksiyonu ile Delege'nin parametre sayısı / türü ve döndürdüğü değerler aynıdır.
- Delege üzerinden New bir degele yaratılıp KareAl fonksiyonuna bağlanır.
- Delege üzerinden KareAl fonksiyonu kullanıma hazırdır.

Multicasting

- Bir delegeyi C veya C++'taki function pointerlerden farklı olarak birden fazla metoda temsilci kılabiliriz. Buna çoklu temsilcilik (Multicasting) denilir.
- Delegenin listesine yeni bir metod eklemek için C#'ta "+=" operatörü kullanılır.
- Listedden bir metod çıkarmak için C#'ta "-=" operatörü kullanılır.
- Bir delegeyle ilişkilendirilmiş metodlar o delegenin çağrı listesinde yer alır. Bu yüzden delegeyle ilişkilendirilmiş metodlara erişmek için delegenin InvocationList() metodu kullanılır.

Multicasting

```
public delegate void MyDelegate(int Deger1, int Deger2);

void Toplama(int Sayi1, int Sayi2)
{
    Console.WriteLine("{0} + {1} = {2}", Sayi1, Sayi2, Sayi1 + Sayi2);
} //Toplama

void Carpma(int Sayi1, int Sayi2)
{
    Console.WriteLine("{0} * {1} = {2}", Sayi1, Sayi2, Sayi1 * Sayi2);
} //Carpma

void Bolme(int Sayi1, int Sayi2)
{
    Console.WriteLine("{0} / {1} = {2}", Sayi1, Sayi2, Sayi1 / Sayi2);
} //Bolme
```

- Bir delege ve delege özelliklerine sahip üç fonksiyon vardır. Fonksiyonlar diğer örnekten farklı olarak static değildir.

Multicasting

```
public Program()  
{  
    //Constructor  
    MyDelegate delege = null;  
    delege += new MyDelegate(Toplama);  
    delege += new MyDelegate(Carpma);  
    delege += new MyDelegate(Bolme);  
  
    int X, Y;  
    X = 18; Y = 3;  
    delege(X, Y);  
    Delegate[] DlgList = delege.GetInvocationList();  
    Console.WriteLine("Delegenin temsil ettiği metodlar :");  
    Console.WriteLine("-----");  
    for (byte i = 0; i < DlgList.Length; i++)  
        Console.WriteLine(DlgList[i].Method.Name);  
  
    delege -= new MyDelegate(Bolme);  
    int X1, Y1;  
    X1 = 30; Y1 = 5;  
    delege(X1, Y1);  
  
    DlgList = delege.GetInvocationList();  
    Console.WriteLine("Delegenin temsil ettiği metodlar :");  
    Console.WriteLine("-----");  
    for (byte i = 0; i < DlgList.Length; i++)  
        Console.WriteLine(DlgList[i].Method.Name);  
}  
}
```

Delegeye metot ekleme

Mevcut metodun özelliklerini çağırma

Delegeye metot çıkarma

- Program sınıfına ait Program Constructor'ı içine multicasting işlemler yapılmıştır.
- GetInvocationList fonksiyonu dizi olarak Delege listesi döndürür. Bu fonksiyon ile delegeye ait metotları görebiliriz. Bu örnekte for döngüsü ile delegenin içerdiği metotların isimleri döndürülmüştür.