

Yanlış İsimlendirmeler

1-Ağustos-2009

Yıllar önceydi. Bir arkadaşımın kodunu inceliyordum. Kodunda *max* ismini verdiği bir değişken vardı. Ben de doğal olarak bu değişkenin en büyük değeri tuttuğunu sanarak kodu anlamlandırmaya çalıştım. Ancak birşeyler tam yerine oturmuyordu bir türlü. Meğerse arkadaşım başlangıçta en büyük değeri bulup kullanırken sonra en küçük değerin bulunup kullanılması gerektiğini düşünerek algoritmasını değiştirmiş. Fakat değişkenin ismini değiştirmemiş. Düşünebiliyor musunuz, bir değişken var, ismi *max* fakat bu aslında en küçük değeri tutuyor :-).

Brian Kernighan “*The Practice Of Programming*” isimli kitabında değişken isimlerinin kısa (*concise*), akılda kalıcı (*memorable*), bilgi verici (*informative*) ve telaffuz edilebilir (*pronounceable*) olması gerektiğini söyler. Fakat belli ki yerli ve yabancı programcıların çoğu bu tavsiyeye yeterince kulak asmıyorlar. Halbuki değişkenlerin kullanım amacına uygun bir biçimde isimlendirilmesi önemli. Çünkü bir değişkene uygunsuz bir isim verildiğinde o değişkenin kullanıldığı kod parçası da tamamen yanlış anlaşılabilir.

Geçen gün *C#* dersinde dosya işlemleri konusunda *Stream* sınıfının *Read* ve *Write* metotlarını anlatırken aynı zamanda *MSDN*’den metotların parametrik yapılarını öğrencilere gösteriyordum. Fakat metotların ikinci parametrelerini anlatırken zorluk çektiğimi farkettim:

```
public abstract int Read(byte[] buffer, int offset, int count)
public abstract void Write(byte[] buffer, int offset, int count)
```

Read metodu üçüncü parametresiyle belirtilen miktarda *byte* değeri birinci parametresiyle belirtilen diziyeye aktarır. *Write* metodu da bunun tersini yapar. Peki buradaki *offset* parametresinin anlamı nedir? Bu parametre aktarımın dizinin kaçınıcı indeksinden itibaren yapılacağını belirtiyor. Fakat parametre değişkeni için *offset* ismi yerine *index* ismi daha açıklayıcı olmaz mıydı? Her ne kadar *offset* sözcüğü görece uzaklık anlamında diziler için de kullanılıyor olsa da bu bağlamda dosya *offset*’i ile karışması mümkün. Bu parametre değişkenine *offset* yerine *index* ismi verilseydi yanlış çağrışımlar engellenebilirdi. Sanırım *Microsoft* programcıları dizinin bir *byte* dizisi olmasından dolayı *offset* sözcüğünü tercih etmiş. Yoksa kütüphanede benzer metotlarda diziler için hep *index* isminin tercih edildiğini görüyoruz.

Yanlış verilmiş isimlerle her yerde karşılaşmak olası. Fakat *Microsoft* sistemlerinde çalışıyorsanız sanki daha bir olası :-). Bu sistemlerde o kadar çok yanlış verilmiş isimlerle karşılaştım ki... Örneğin, *GetDlgItem* isimli bir *API* fonksiyonu vardır. İsmine bakarsanız bunun diyalog pencerelerine ilişkin bir fonksiyon olduğunu sanırsınız. Fakat aslında bu fonksiyon üst pencerenin *handle* değerini ve alt pencerenin *id* değerini alarak alt pencerenin *handle* değerini verir. Yani aslında çok genel bir fonksiyondur. Buradaki *Dlg* de *Item* da yanlış çağrışımlar yapıyor. Fonksiyonun *GetChildWnd* gibi bir isme sahip olması daha uygun olmaz mıydı? Peki, *IsBackground* gibi bir *property* ismi sizde ne çağrışım uyandırıyor? İsim soru eki içerdiğine göre bunun *read only* bir *property* olması gerektiğini düşünürsünüz değil

mi? Yok işte, .NET'te *Thread* sınıfının bu *property*'si *thread*'i arka plan *thread* yapmak için kullanılabilen *read/write* bir *property*'dir:

```
thread.IsBackground = true;
```

ExecuteNonQuery isimli veritabanı metodunun ne yaptığını sanırsınız? İsmine bakılırsa sorgulamaya yol açmayan işlemleri yapıyor olmalı. Metodu yazanın sorgulama (*query*) kavramı konusunda kafası biraz karışmış galiba. Aslında veritabanı kayıtları üzerinde yapılan her türlü işleme sorgulama deniyor. Bir *SQL* komutunun sorgulama olması için bize kayıt döndürmesi gerekmiyor. Burada ismi veren kişi muhtemelen işlemin sorgulama olması için kayıt geri döndürmesi gerektiğini sanmış...

Yanlış isimlendirmelerle yalnızca *Microsoft* dünyasında karşılaşmış değilim. Aslında kafamda burada yazdıklarımın dışında daha pek çok yanlış isimlendirme örnekleri var. Siz de karşılaşmış olduğunuz yanlış isimlendirme örneklerini bana e-posta yoluyla bildirirseniz sevinirim. Bunların geniş bir listesini çıkartmak istiyorum...