



Özet

•İSİM UZAYI HAZIRLAMA VE KULLANMA

•Çoğu programlama dilinde, programcılarını işini kolaylaştırmak için hazır kütüphaneler bulunmaktadır. C# programlama dilinde ise herhangi bir hazır kütüphane bulunmamaktadır. Bunun yerine .NET Framework'ün sunduğu sınıf ve tür olarak adlandırılan yapılar bulunmaktadır. Bu yapılar sayesinde temel programlama işlemleri programcılar tarafından kolayca gerçekleştirilmektedir. .NET Framework sınıf kütüphaneleri kod yazarken programcıya fayda sağlayacak birçok servisi, nesneyi, yapıları ve benzerlerini içermektedir. Programcının bu sınıfları kolayca kullanabilmesi için sınıflar mantıksal bir organizasyonla sunulmaktadır. Bu mantıksal organizasyonu sağlayan ise *isim uzayları (namespace)*'dir.

•İsim uzayı (namespace), benzer amaca hizmet eden sınıfların bir arada tutulmasıyla oluşan bir küme olarak tanımlanabilir. Bu küme içinde sınıfların yanı sıra başka isim alanları (namespace) veya yapılar da bulunabilir. İsim uzayları ile efektif programlar yazılmaktadır. Özellikle büyük programlar çok fazla sınıf, metod gibi yapılar içerdiği için bu programların bakımı ve okunabilirliği oldukça zordur. İsim uzayları bu gibi problemleri çözerek programcıya kolaylık sağlamaktadır.

•İSİM UZAYI (NAMESPACE) NEDİR?

•Birden fazla kişi tarafından geliştirilen projelerde programcılar, geliştirme aşamasında sınıf, tür, değişken gibi yapıları aynı isimler verebilir. Bu da projede isim çakışmalarına neden olmaktadır. Modern programla dillerinde bu sorunu çözmek için mantıksal organizasyon yapısına başvurulmuştur. C# ta programlama içerisindeki bu mantıksal organizasyon isim uzayları (namespace) ile sağlanmaktadır.

•İsim Uzaıy Bildirimi

•İsim uzayları içindeki yapıları erişebilmek için ya kod içerisinde isim uzayı ve kullanılacak yapının tam adı yazılır ya da programın başında kullanılacak isim uzayı *using* anahtar kelimesi ile çağrılır. İsim uzayı içerisinde sınıf (class), yapı (struct), numaralandırma (enum), temsilci (delegate) ya da ara yüz (interface) bildirimi yapılabilirken değişken veya metod bildirimi yapılamaz.

•İsim Uzaıy Oluşturma

•Program içerisinde okunabilirliği artırmak ve kullanımı kolaylaştırmak için programcılar kendi isim uzaylarını oluşturabilir. İsim uzayı *namespace* anahtar kelimesi kullanılarak oluşturulur. İsim uzayının adı belirtildikten sonra isim uzayı içerisinde yer alacak işlemler tanımlanır. Daha sonra isim uzayı program içerisine dahil edilerek kullanımı sağlanır.

•İç İçe İsim Uzaıy Tanımlama

•C# programlama dilinde iç içe birden fazla isim uzayı tanımlanabilir. Böylece kod içerisinde hiyerarşik bir düzen sağlanabilmektedir. İç içe isim uzayının tanımlandığı durumda isim uzaylarına "." operatörü ile erişilir. İç içe isim uzaylarını koda eklemek için *using* anahtar kelimesi ile birlikte isim uzay adının tam yolu yazılmalıdır. Tam yol yazılırken de yine isim uzay adları arasına "." operatörü eklenmelidir.

•Programda Kullanılan İsim Uzaıylarının Listesi

•C# programlama dilinde kod içerisinde kullanılan isim uzayları Visual Studio'da yer alan View menüsü altındaki Object Browser'da listelenmektedir. Ayrıca Solution Explorer penceresinde bulunan References altında da görülebilir.

•C# programlama dilinde yaygın kullanılan birkaç isim uzayı ve özellikleri;

•Data

•Çoğunlukla veri tabanı yer alan projelerde kullanılan bir isim uzayıdır. Bu isim uzayı içerisinde yer alan veri tipleri ile veri tabanına bağlantı kurulması sağlanır.



Özet (devamı)

•System

•Bu isim uzayı altında generic koleksiyonlar, ortak kullanıma sahip değerler, çeşitli ara yüzler, referans edilen veri türleri ve işlem hatalarını tanımlayan temel sınıflar yer alır.

•Drawing

•Bu isim uzayı altında font ayarları, renklendirme, gibi temel grafik işlemlerinin kolayca yapılabilmesini sağlayan sınıflar yer almaktadır.

•Configuration

•Bu isim uzayında, programlar için .Net Framework konfigürasyon ayarlarının yapılmasını sağlayan hazır ara yüz ve sınıflar yer alır.

•IO

•Bu isim uzayı ile dosya okuma ve yazma işlemlerinin yapılmasını sağlayan sınıflara erişilebilmektedir.

•Text

•Bu isim uzayı ASCII ve Unicode karakter kodlarını temsil eden, karakter ve bayt dönüşümü yapan sınıfları içermektedir. Bu isim uzayı kullanılarak şifreleme, şifre çözme ve text işlemleri gerçekleştirilebilir.

•Web

•Bu isim uzayı ile istemci/sunucu iletişimini sağlayan sınıf ve ara yüzlere erişilebilmektedir.

•Threading

•Uygulamaların aynı anda paralel işlemler yapmasını sağlayan sınıflar için bu isim uzayı kullanılmaktadır.

•Net

•Bu isim uzayında ağlarda kullanılan birçok protokol için hazır sınıflar yer almaktadır.

•Globalization

•Uygulamalarda kullanılan dil, tarih, rakam, takvim ve para birimi gibi bilgilerin tanımlanmasını sağlayan sınıflar bu isim uzayında yer almaktadır.

•Security

•Bu isim uzayında güvenlik ve yetkilendirme işlemleri için hazır sınıflar bulunmaktadır.

•UI

•Kullanıcı ara yüzünü ve web sayfalarını oluşturmak için gerekli sınıf ve ara yüzler bu isim uzayında yer almaktadır.

DEĞERLENDİRME SORULARI

- C# programlama dilinde isim uzayı oluşturmak için aşağıdaki anahtar kelimelerden hangisi kullanılır?
 - Create
 - System
 - Import
 - Namespace
 - Data
- Aşağıdaki yapılardan hangisinin bildirimi isim uzayı içerisinde yapılamaz?
 - Metot
 - Sınıf
 - Numaralandırma
 - Ara yüz
 - Temsilci
- C# programlama dilinde veri tabanı ile ilgili işlemleri yapabilmek için aşağıdaki isim uzaylarından hangisi kullanılır?
 - UI
 - Security
 - Data
 - System.Net
 - Web
- İsim uzayları ile ilgili verilen bilgilerden hangisi yanlıştır?
 - C# programlama dilinde iç içe isim uzayı tanımlanabilir.
 - İsim uzayı içerisinde sınıf bildirimi yapılabilir.
 - Dosya okuma ve yazma işlemleri için Data isim uzayı kullanılır.
 - Okunabilirlik açısından isim uzay isimlerinin kısa olması tercih edilir.
 - Farklı dosyalarda aynı isim uzayları yer alabilir.
- Aşağıdaki anahtar kelimelerden hangisi kullanılarak isim uzayındaki sınıf ve yapılara erişim sağlanır?
 - Create
 - Using
 - File
 - Metot
 - Import

- C# programlama dilinde isim uzayına ait yapılara aşağıdaki operatörlerden hangisi ile erişilir?
 - “,” operatörü ile;
 - “!” operatörü ile;
 - “*” operatörü ile;
 - “-” operatörü ile
 - “.” Operatörü ile;
- “Universite” adlı isim uzayı içerisinde “Ogrenci” adlı ikinci bir isim uzayının oluşturulduğu durum için aşağıdaki ifadelerden hangisi kullanılarak “Ogrenci” isim uzayı koda eklenir?
 - using Universite;
 - using System.Universite;
 - using Universte.Ogrenci;
 - using Ogrenci;
 - using System.Universite.Ogrenci
- Aşağıdakilerden hangisi isim uzayı tanımlarken dikkat edilmesi gereken kurallardan biri değildir?
 - İsim uzay adlarında büyük harf kullanılmaz.
 - İsim uzay adları içeriğindeki yapı ile ilişkili olmalı.
 - İsim uzaylarında metot veya değişken bildirimi yapılmaz.
 - İsim uzay adlarının çok uzun olmamasına özen gösterilir.
 - İsim uzayında ara yüz bildirimi yapılabilir.
- Aşağıdaki isim uzaylarından hangisi “Brush”, “Font”, “Color” gibi sınıfları barındırır?
 - Data
 - Security
 - Globalization
 - UI
 - Drawing
- C# programlama dilinde sınıfların mantıksal organizasyonu aşağıdaki yapılardan hangisi ile sağlanır?
 - Metot
 - Değişken
 - Numaralandırma
 - İsim Uzayı
 - Temsilci

Cevap Anahtarı

1.d, 2.a, 3.c, 4.c, 5.b, 6.e, 7.c, 8.a, 9.e, 10.d