

ANKARA ÜNİVERSİTESİ GAMA MESLEK YÜKSEKOKULU

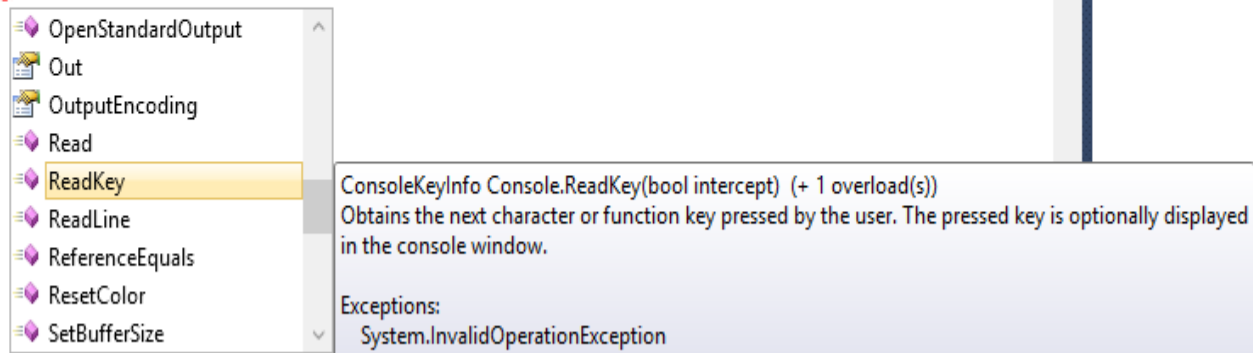
* BMT255 ALGORİTMA VE PROGRAMLAMA

* Öğr.Gör.Uğur YEDEKÇİOĞLU

* **IntelliSense kullanarak kod yazma**

- * IntelliSense, nesne tabanlı programlama yazılımının kod tamamlama özelliğidir. Bu özellik daha hızlı ve hatasız kod yazma olanağı sağlar. Bir intellisense önerisini kabul etmek için açılan listeden ok tuşları yardımıyla seçtikten sonra Tab, Enter ya da boşluk tuşlarından birisine basılarak kullanılabilir. Bu işlem fare ile de yapılabilir.
- * **NOT:** Herhangi bir durumda intellisense öneri listesini görmek için **Ctrl + Boşluk** tuş kombinasyonu kullanılabilir.
- * `Console.WriteLine` ifadesini yazarken **Intellisense** özelliğinden yararlanarak nasıl yazıldığı Resim aşağıda görülmektedir.

```
static void Main(string[] args)
{
    Console.WriteLine("BMT255 Algoritma ve Programlama");
    Console.ReadKey();
    Console.
}
```



The screenshot shows an IDE's autocomplete menu for the `Console` class. The `ReadKey` method is selected and highlighted in yellow. A tooltip is displayed to the right of the menu, providing details about the `ReadKey` method.

Method	Description
<code>OpenStandardOutput</code>	
<code>Out</code>	
<code>OutputEncoding</code>	
<code>Read</code>	
<code>ReadKey</code>	<code>ConsoleKeyInfo Console.ReadKey(bool intercept) (+ 1 overload(s))</code> Obtains the next character or function key pressed by the user. The pressed key is optionally displayed in the console window.
<code>ReadLine</code>	
<code>ReferenceEquals</code>	
<code>ResetColor</code>	
<code>SetBufferSize</code>	

Exceptions:
System.InvalidOperationException

* **Namespace**

- * Namespaceler(isim uzayları) nesneye dayalı, hiyerarşik ve birleşik, içinde çok sayıda sınıf, arayüz ve yapı bulundurabilen kütüphaneler olarak tanımlanabilir.
- * İsim uzayları program yazmanın en önemli parçasıdır ve daha çok yazılan kodların tekrar kullanılabilirliğini artırmak için geliştirilmiştir. Yazılan kodlarla program büyüdükçe iki sorun ortaya çıkar. Birincisi büyük programları anlamak ve bakımını yapmak küçük program parçacıklarına göre çok daha zordur. İkincisi ise daha fazla kod, ad, metot, sınıf demektir. Programda kullanılan ad sayısı arttıkça iki ya da daha fazla adın çakışma ihtimali, dolayısıyla projenin hata verme olasılığı da artar.

- * İsim uzayları(ad alanları) bu problemlerin çözümünde bize büyük kolaylıklar sağlarlar. Örneğin, aynı adla iki sınıf, farklı isim uzaylarında yer alıyorsa birbirine karıştırılmaz.

* Using İfadesi

- * Tüm sınıfların bir isim uzayı içerisinde yer almasının sağladığı kolaylıklardan, bir önceki konuda söz edilmişti. Nesne tabanlı programlama yazılım ortamının etkin bir şekilde kullandığı yazılım geliştirme paketi .NET Platformu içinde her bir sınıf, bir ad alanı içinde yer alır. Örneğin, **Console** sınıfı **System** ad alanı içerisinde yer alır. Bu, sınıfın tam adının aslında **System.Console** olması demektir.

* **Grafiksel Arayüz**

- * Şimdiye kadar basit konsol uygulaması oluşturuldu ve çalıştırıldı. Nesne tabanlı programlama yazılımı ortamı, aynı zamanda grafiksel windows tabanlı uygulamalar oluşturmak için ihtiyaç duyulan her şeyi kapsar. Windows tabanlı uygulamanın forma dayalı kullanıcı arayüzü etkileşimli olarak tasarlanabilir.

```
using System;  
using System.Collections.Generic;  
using System.Linq;  
using System.Text;
```


Kaynakça

<http://immibilisim.com/moduller/1-%20Nesne%20Tabanlı%20Programlamaya%20Giriş.pdf>