

Bu doküman boyunca Visual Studio'yu kullanarak nasıl bir yazılım projesi oluşturabileceğinizi öğreneceksiniz. İnternette ve yapay zekadan yardım almayı unutmayın, ancak her şeyi kendiniz yazmayı deneyin. **Kopyala yapıştır yapmaktan kaçının!** Bu şekilde daha hızlı öğrenirsiniz.

# Yazılıma Giriş

## İlk Projeyi Oluşturma

Oğuzhan Karagüzel

19.10.2024

## İçindekiler

Şekiller Dizini .....	2
GELİŞTİRME ORTAMINI TANIMA.....	3
Visual Studio'yu Tanıma.....	3
Proje Oluşturma .....	3
İlk Proje.....	7
Solution Explorer .....	7
View Menüsü.....	8
Output (Çıktı) ve Error List (Hata listesi) Penceresi.....	9
Ana Ekran ya da Dosya Düzenleme Ekranı.....	10
İlk projeyi çalıştırma.....	10
ÖDEVLER.....	13
TEST .....	14
Cevap Anahtarı .....	14

## Şekiller Dizini

Şekil 1 "Visual Studio" Açılış Penceresi .....	3
Şekil 2 Yeni Proje Taslak Seçimi.....	4
Şekil 3 Projeyi ve Çözümü Adlandırmak .....	5
Şekil 4 Proje ve Çözümü Kurallar Dahilinde Adlandırma .....	6
Şekil 5 İlk Proje.....	7
Şekil 6 View Menüsü .....	9
Şekil 7 Hata mesajı.....	10
Şekil 8 İlk Projeyi Çalıştırma .....	11
Şekil 9 Projenin Çalıştırılması.....	11
Şekil 10. ilk kod denemesi .....	12
Şekil 11 İlk Kodu Çalıştırma.....	13

Öğuzhan KARAGÜZEL

## GELİŞTİRME ORTAMINI TANIMA

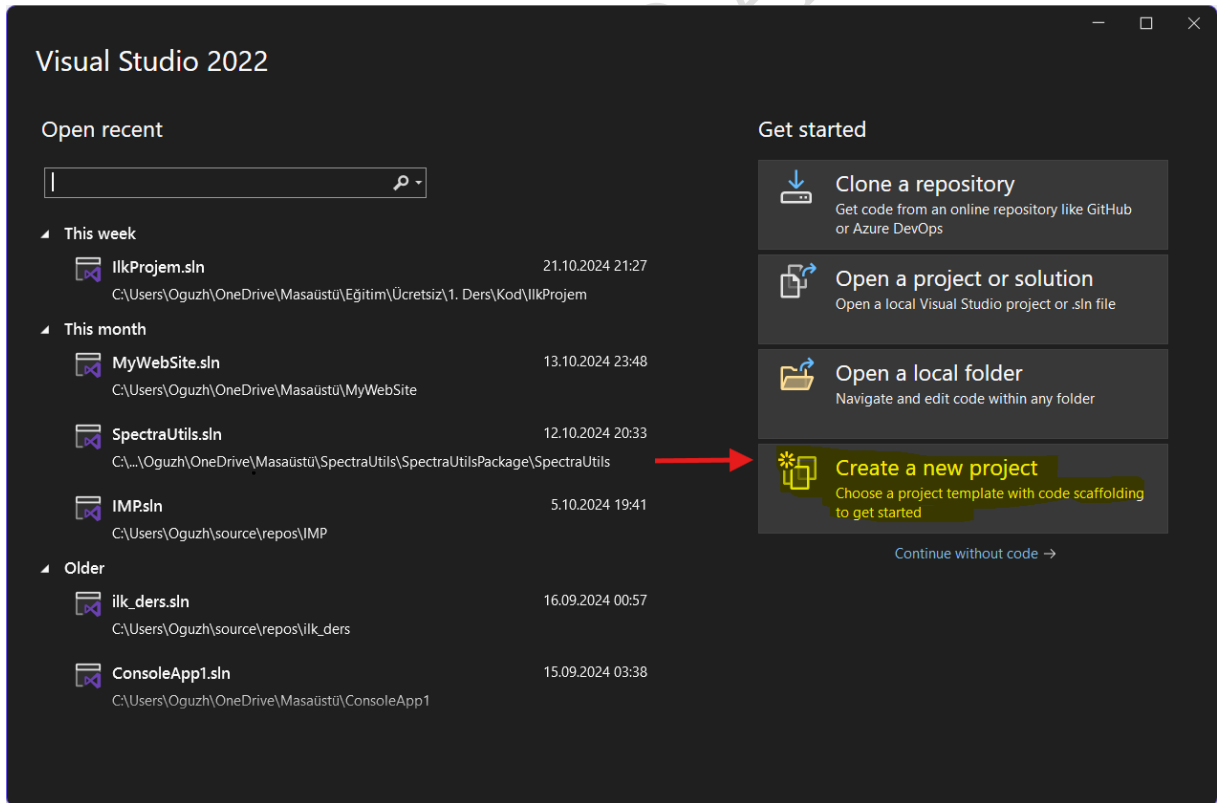
### Visual Studio'yu Tanıma

Eğitim boyunca kullanacağımız araç, **Visual Studio** olacak. Projeleri oluşturmak ve geliştirmek için bu aracı kullanacağız. İlk başlarda bazı detaylara derinlemesine girmeyeceğiz, çünkü deneme-yanılma yoluyla keşfetmek öğrenmeyi hızlandıracaktır.

### Proje Oluşturma

**Visual Studio'yu açın.** Eğer yüklü değilse, indirip kurmanız gerekiyor. Bunu Youtube eğitim videolarının ilkinizi izleyerek nasıl yapabileceğinizi öğrenebilirsiniz. Açtıktan sonra karşınıza "Visual Studio" açılış penceresi gelecek (Şekil 1).

"Visual Studio" programına çift tıklayarak şekil 1'de ki pencereye ulaşın.



Şekil 1 "Visual Studio" Açılış Penceresi

Bu pencere daha önce açmış olduğunuz projeleri sol tarafta görebilirsiniz. Sağ tarafta ise size farklı seçenekler sunulmaktadır. Kısaca bu seçenekleri açıklayalım;

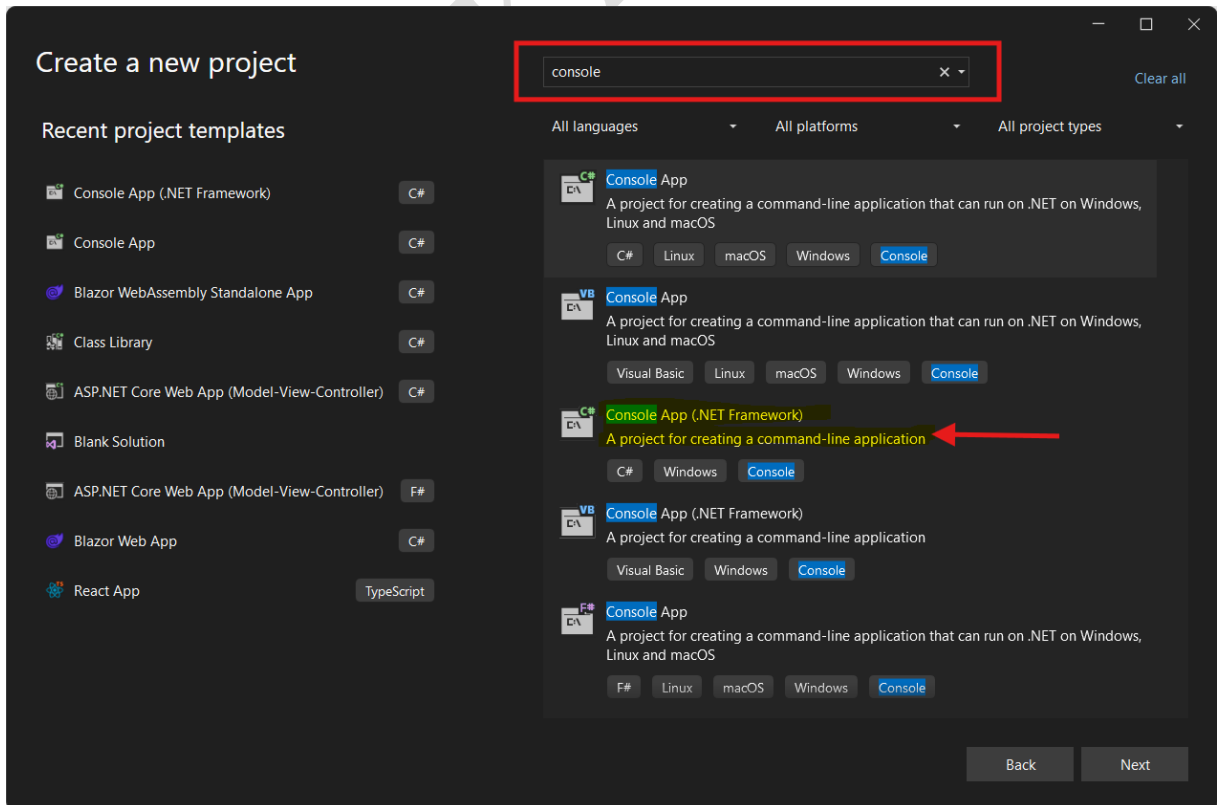
- **Clone a Repository** : Bu aşamada git kullanımını henüz bilmiyorsunuz. Bundan dolayı bu seçenekle zaten hiç işiniz olmayacak. Benzetme yaparak kısaca açıklama gerekirse. İnternete yüklenmiş kodları bilgisayarınıza doğrudan proje olarak indirip açmanızı sağlaya seçenek denilebilir. Bunu aklınızda bir şeylerin canlanması için söylüyorum. Bu dediklerim bir benzetmedir. Bundan dolayı hatalıdır ve eksiktir.
- **Open a Project or solution** : Bu seçeneğe tıkladığınızda dosya seçim ekranı açılacaktır. Burada bilgisayarınızda var olan bir projeyi seçmeniz beklenmektedir. Bu projeyi doğrudan açmış olacaksınız. Burada ki “Solution” kavramına daha sonra değinilecektir.
- **Open a local folder** : Bu seçenekte ise içinde projelerin bulunduğu bir klasörü tümünden seçmenizi sağlayan bir seçenek diyebilirim. Seçtiğiniz klasörü tamamiyle açar. Özellikle bir projeyi ya da solution’u açmazsınız.
- **Create a new Project** : Burada doğrudan yeni bir proje oluşturursunuz. Eğitim süresince kullanacağınız seçenek budur. Ya da şekil 1’de görüldüğü gibi sol tarafta daha önce çalıştığınız projelere tıklayarak doğrudan açabilirsiniz.

Bizim işimiz ise yeni bir proje açarak başlayacak. Lâkin öncesinde “solution” kavramından bahsetmek gerekir.

İlerleyen süreçte göreceksiniz ki bir proje geliştirdiğinizde ya da geliştirirken birden fazla projeye aynı anda ihtiyaç duyacaksınız. Bundan dolayı birden fazla projeyi aynı çatı altında bulundurduğunuz yapıya “Solution” yani çözüm denmektedir. (Burada yine bir benzetme yapılmış olup bilgiler hatalı ve eksiktir.) .sln uzantısına sahip özel bir dosyadır. Genel olarak bu dosya ile bir işiniz olmayacaktır. Sadece çözümünüzü temsil ettiğini düşünmeniz yeterlidir.

Hadi o zaman ilk projemizi oluşturalım;

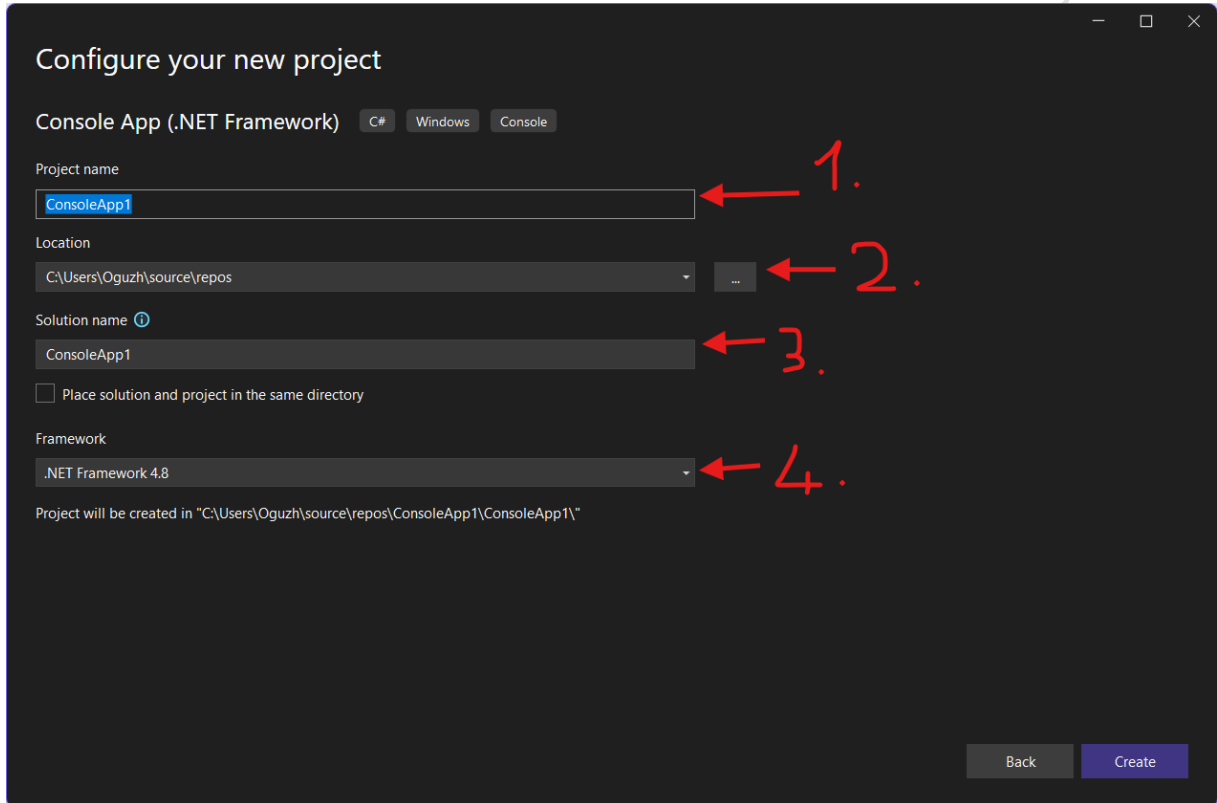
Şekil 1. de kırmızı ok ile gösterilen “Create a new project” seçeneğine tıklayalım.



Şekil 2 Yeni Proje Taslak Seçimi

Şekil 2’de ki pencereye ulaşacaksınız. Bu pencerede sol tarafta daha önce kullandığınız proje taslakları sıralanacaktır. Hızlı ulaşmanız için “Visual Studio” tarafından size gösterilir. Sağ tarafta ise bütün taslaklar sıralanmaktadır. Burada ki taslaklardan bir tanesini seçerek bu aşamayı geçebilirsiniz. Yukarıda ki kırmızı dikdörtgen içerisine alınmış alan “Search” yani arama alanıdır. Buradan aradığınız taslağın adını girerek taslaklar içerisinde filtreleme yapabilirsiniz.

Bu arama alanına “Console” yazarak filtreleme yapınız. Ardından şekil 2’de kırmızı ok ile gösterilen seçeneği bulunuz ve seçiniz. Eğitim sürecinin başında .Net framework ve “console” uygulaması ile başlayacağız. Bunda dolayı bu seçeneği seçmeniz gereklidir. Seçeneğin altında etiketler göreceksiniz. Bu etiketlerden C# olan projeyi seçiniz.



Şekil 3 Projeyi ve Çözümü Adlandırma

Şekil 3’deki pencereye ulaşacaksınız. Şekil 3’de görülen numaralara göre açıklama yaparsak;

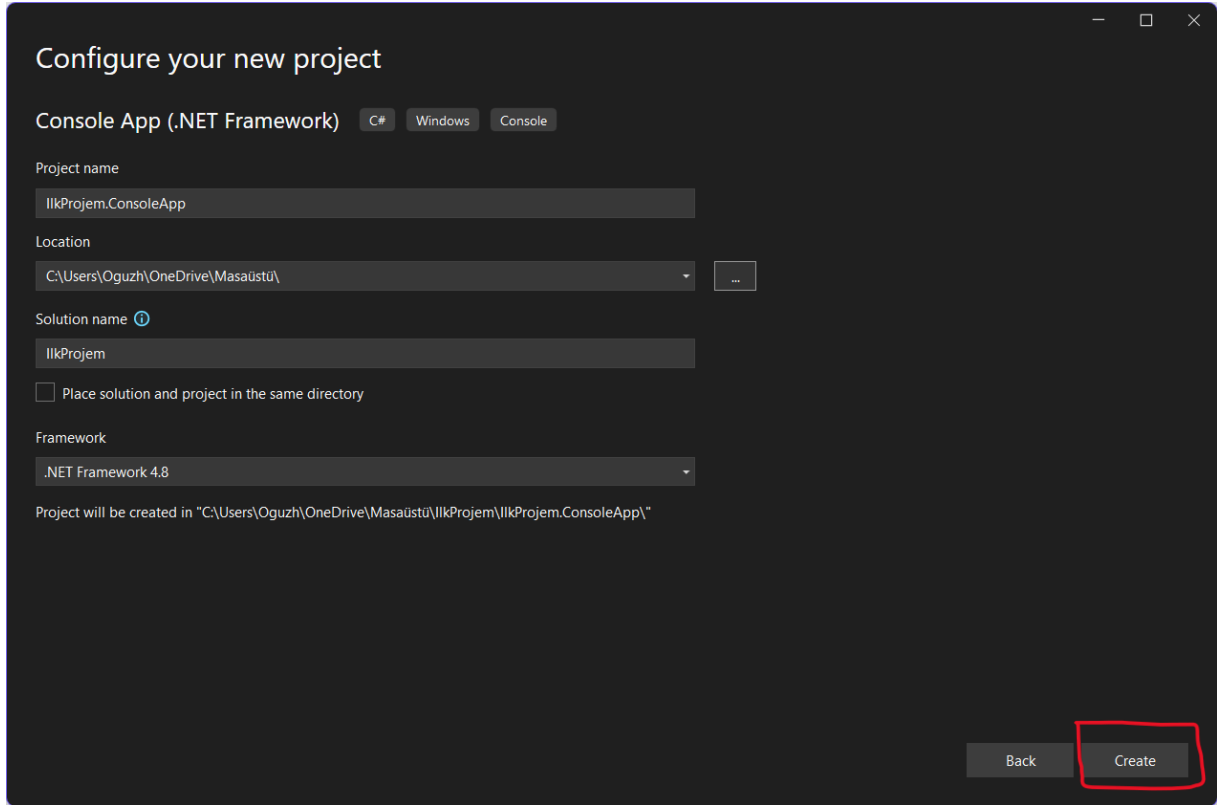
1. Projenizin adı.
2. Projenizin oluşturulacağı dosya konumu.
3. Çözümünüzün adı.
4. Projenizin sürümü.

Projenizi ve çözümünüzü adlandırırken dilediğiniz gibi adlandıramazsınız. Bazı kurallara uymayı gerekmektedir. Bu kurallar şu şekildedir;

- Proje ya da çözüm adı Türkçe karakter içermemelidir.
- Proje ya da çözüm adı boşluk içermemelidir. Boşluk yerine nokta (.) ya da (\_) kullanabilirsiniz.
- Çözüm adı geneli adlandırırken projeniz kendi içeriğine göre adlandırılır. Bu kurala şu anda uymanıza gerek yoktur. Buna örnek verilecektir. Ancak eğitimin ilk aşamalarında tek projeniz olduğunuzdan çözüm ile proje aynı ad ile oluşturulmasında hiçbir sakınca yoktur.

- Projenin sürümü ihtiyaca göre seçilir. Ancak yeni açılan projelerde genellikle en son sürüm ya da en son çift sayı ile başlayan sürüm seçilir. Sürüm ve sayıları hakkında ileride detaylı bilgi verilecektir.

Hadi ilk projemizi adlandıralım. Yukarıdaki kurallara uyarak projenizi ve çözümünüzü adlandırabilirsiniz. Ya da şekil 4’teki gibi yaptığımız yapabilirsiniz.



Configure your new project

Console App (.NET Framework) C# Windows Console

Project name  
IlkProjem.ConsoleApp

Location  
C:\Users\Oguzh\OneDrive\Masaüstü\

Solution name ⓘ  
IlkProjem

☐ Place solution and project in the same directory

Framework  
.NET Framework 4.8

Project will be created in "C:\Users\Oguzh\OneDrive\Masaüstü\IlkProjem\IlkProjem.ConsoleApp\"

Back Create

Şekil 4 Proje ve Çözümü Kurallar Dahilinde Adlandırma

Proje adı : IlkProjem.ConsoleApp

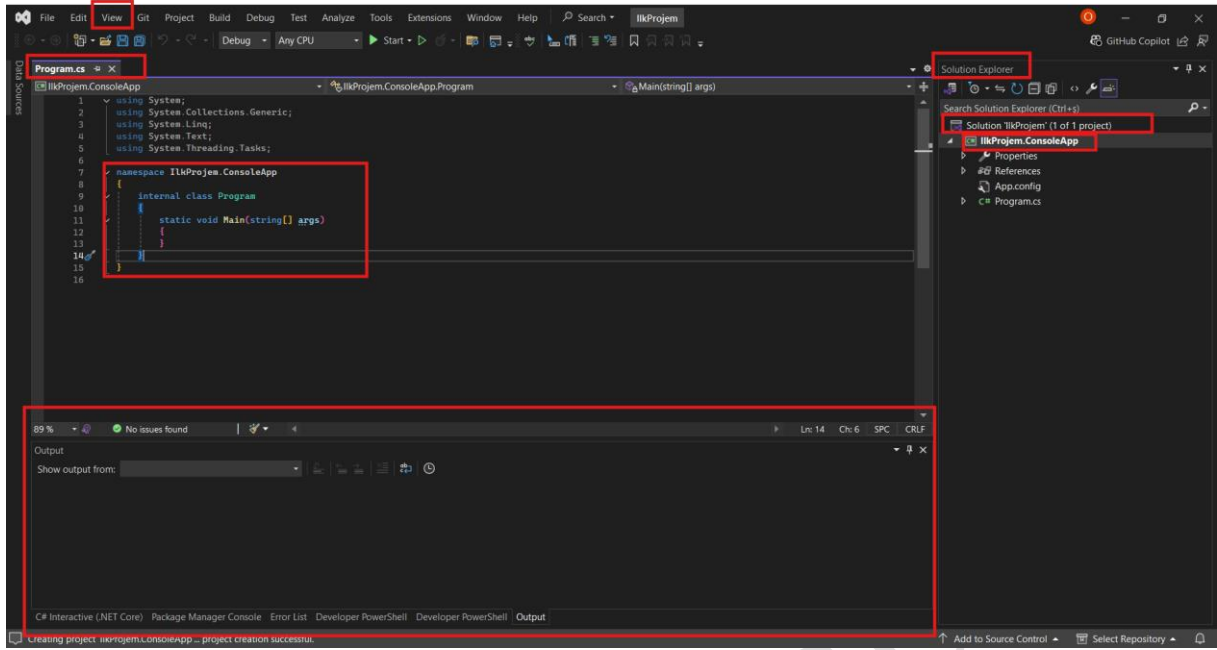
Lokasyon : Masaüstü

Çözüm adı : IlkProjem

Sürüm : 4.8

Sizde bu seçenekleri uygularsanız eğitim süreci boyunca senkronize ilerleriz ve bir hata ile karşılaşmazsınız.

Adlandırmayı tamamladıktan sonra Şekil 4’de kırmızı dikdörtgen içine alınmış “Create” tuşuna basarak ilk projenizi oluşturabilirsiniz.



Şekil 5 İlk Proje

## İlk Proje

Şekil 5’de ki pencere ulaştığınızda ilk proje ekranına ulaşmış olacaksınız. Bu ekranda dikkat etmeniz gereken her şey şekil 5’te dikdörtgen içerisine alınmıştır. Şimdi bunları adım adım açalım.

### Solution Explorer

Bir proje oluşturduğunuzda, projenin tüm dosyalarını ve yapılarını yönetmek için **Solution Explorer** penceresini kullanacağız. Bu pencere tüm projelerinizin bir haritası gibi çalışır.

“Solution Explorer” yani Çözüm gezgini, çözümümüze eklediğimiz tüm dosyalara ve projelere tek tık ile ulaşabilmemizi sağlayan bir araçtır. Bu Pencere sizin için en önemli ve en çok kullanacağınız ikinci penceredir. Daha öncede belirttiğim gibi çözümünüzde birden fazla proje olacaktır. Çözüm gezgininde en üsteki dosyaya dikkat ediniz. Solution ‘İlkProjem’ (1 of 1 project). Gördüğünüz gibi çözümümüz 1 proje içermektedir. Adı da bizim belirlediğimiz “İlkProjem”dir.

Şimdi ise ikinci dosyaya bakın. “İlkProjem.ConsoleApp”. Tam da adlandırdığımız gibi konsol uygulamamız çözüm içerisinde.

Buradaki dosyalar içerisinde tek tıkla gezinebilirsiniz. Bizim ilk aşamada asıl ilgileneceğimiz .cs uzantılı olan (cSharp – Si Şarp) dosyalar. Burada Sadece Program.cs Dosyamız bulunmaktadır. İleride daha fazla dosya açacağız.

Ancak Program.cs Dosyasına kısaca değinmek gerekmektedir. Hangi projeyi açarsanız açın program.cs dosyası ve onun içerisinde yer alan bir “Main” kod bloğu bulunacaktır. Bu dosya ve kod blokları özeldir. Hangi projeyi geliştirseniz geliştirin projeniz her zaman program.cs dosyasının içerisinde ki “main” kod bloğundan başlar. Bu değişmez bir kuraldır. Çünkü programınızın bir giriş noktasının ya da başlangıç noktasının olması gerekir. Bizim başlangıç noktamız burası olacaktır. Kodlarımızı “Main” içerisine yazacağız. Şimdilik bunu es geçebiliriz. İleride bu konuya daha detaylı olarak değinilecektir. Kısaca :

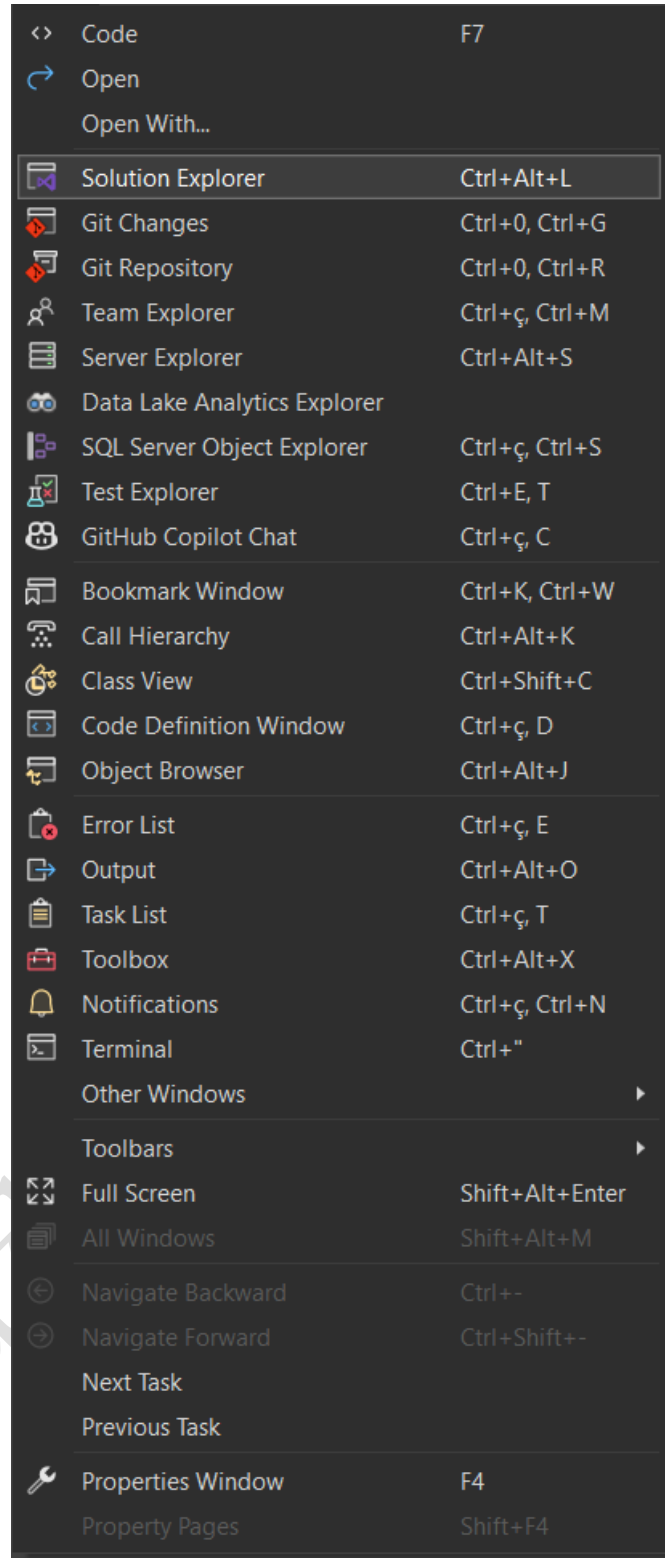


Projeyi oluřturduktan sonra, projede **Program.cs** adında bir dosya göreceksiniz. Bu dosyanın içinde **Main** metodu yer alır. **Main metodu**, programın başladığı noktadır. Her C# projesi, bu metotla başlar. Yani, yazdığınız kodlar ilk olarak burada çalışır.

## View Menüsü

Varsayalım ki “Solution Explorer” penceresini kapattınız. Peki nasıl açacaksınız. Tabi ki de “Ctrl + Alt + L” tuş kombinasyonu ile. Yada daha sonrada çok kullanacağınız view menüsünü kullanarak yapabilirsiniz.

Oğuzhan KARAGÜZEL

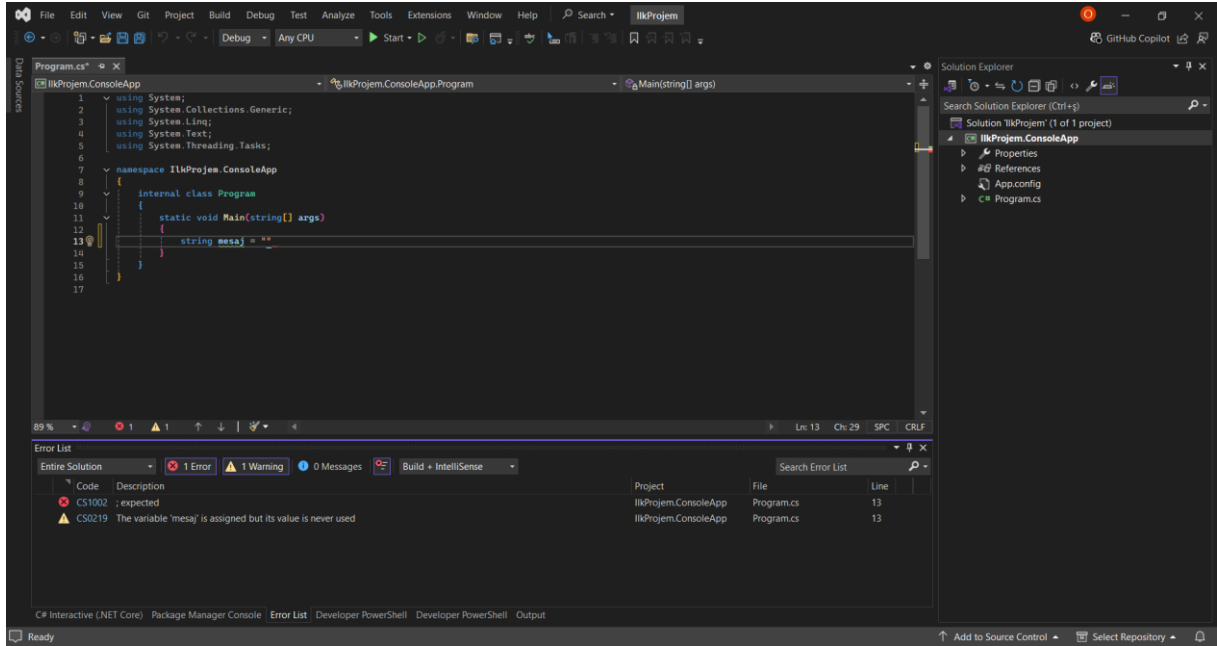


Şekil 6 View Menüsü

Bu menüden çözüm gezgininizi Şekil 6'da gördüğünüz gibi tek tıkla açabilirsiniz.

Output (Çıktı) ve Error List (Hata listesi) Penceresi

Şekil 6'da en altta bulunan dikdörtgene dikkat edin. Bu pencerenin altında pek çok (tab) sekme bulunmaktadır. Şimdilik bizi en çok ilgilendiren Error list'dir. Error list penceresi projemizin yazımı sırasında bir hata yaparsak bize ayrıntıları gösteren penceredir. Şekil 7'de örneğini görebilirsiniz.



Şekil 7 Hata mesajı

Kod yazarken sık sık hata yapacaksınız, bu çok normaldir! **Error List** penceresi, projede karşılaştığınız hataları gösterir. Örneğin, ; **expected** hatası, satır sonunda noktalı virgül (;) koymayı unuttuğunuzda ortaya çıkar.

Gördüğünüz gibi hata penceresinde yaptığım hatayı anında görebiliyorum. Konum ve satır bilgisi ile. Program.cs dosyasında 13. Satırda “;” koymayı unutmuşum. Hemen uyarıldım! Bunun bir yazılımcı için nasıl önemli bir araç olduğunu çok daha ileride anlayacaksınız. Ancak şimdilik kısa kesiyorum.

Output penceresi ise şu an için önemli değildir. Ancak ileride işinize fazlaca yarayacaktır. Bundan dolayı şimdiden aşına olmanız sizin için iyidir.

Ana Ekran ya da Dosya Düzenleme Ekranı.

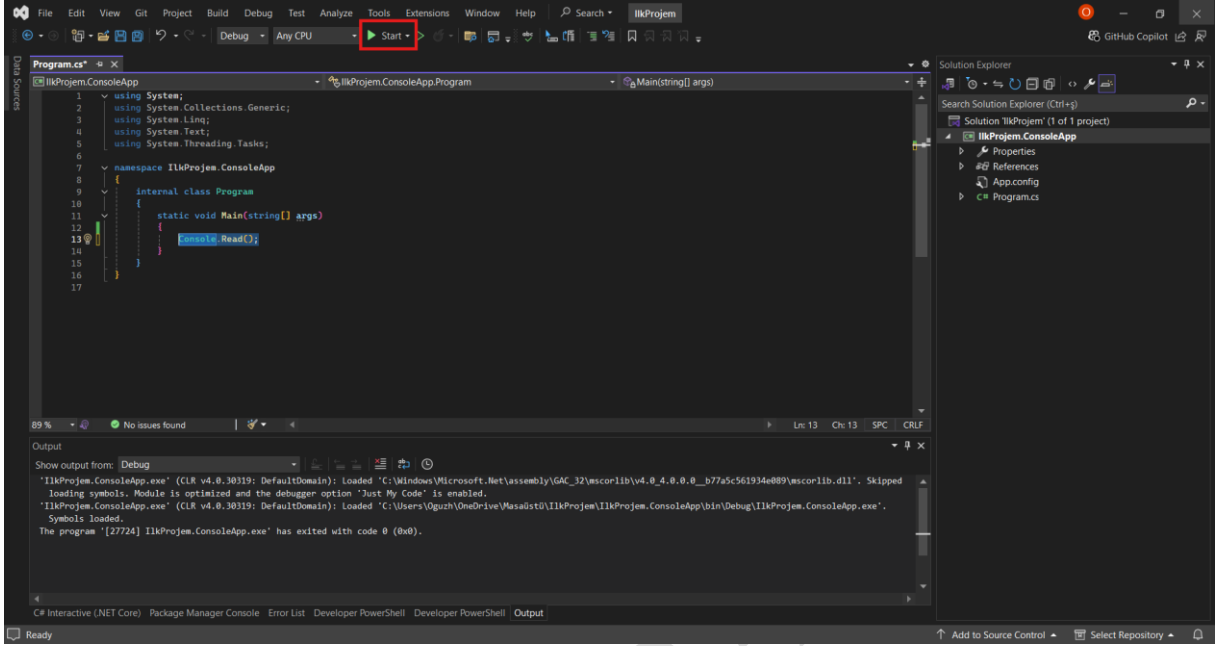
Ekranın tam ortasında bir Word dosyası gibi dosyamızı düzenlediğimiz pencere bulunmaktadır. En çok vakit geçireceğimiz araç budur. Burada gördüğünüz üzere program.cs dosyası açıktır ve yukarıda bir sekme olarak görülmektedir. Burada birden fazla dosya açabiliriz. Bunlara sekmeler aracılığı ile geçiş yapabiliriz. Burası amacını oldukça belli etmektedir. Bundan dolayı üzerinde fazla durmaya gerek yoktur.

İlk projeyi çalıştırma.

Şekil 8'de gösterildiği gibi main kod bloğunun içerisine lütfen şu kod bloğunu yazınız.

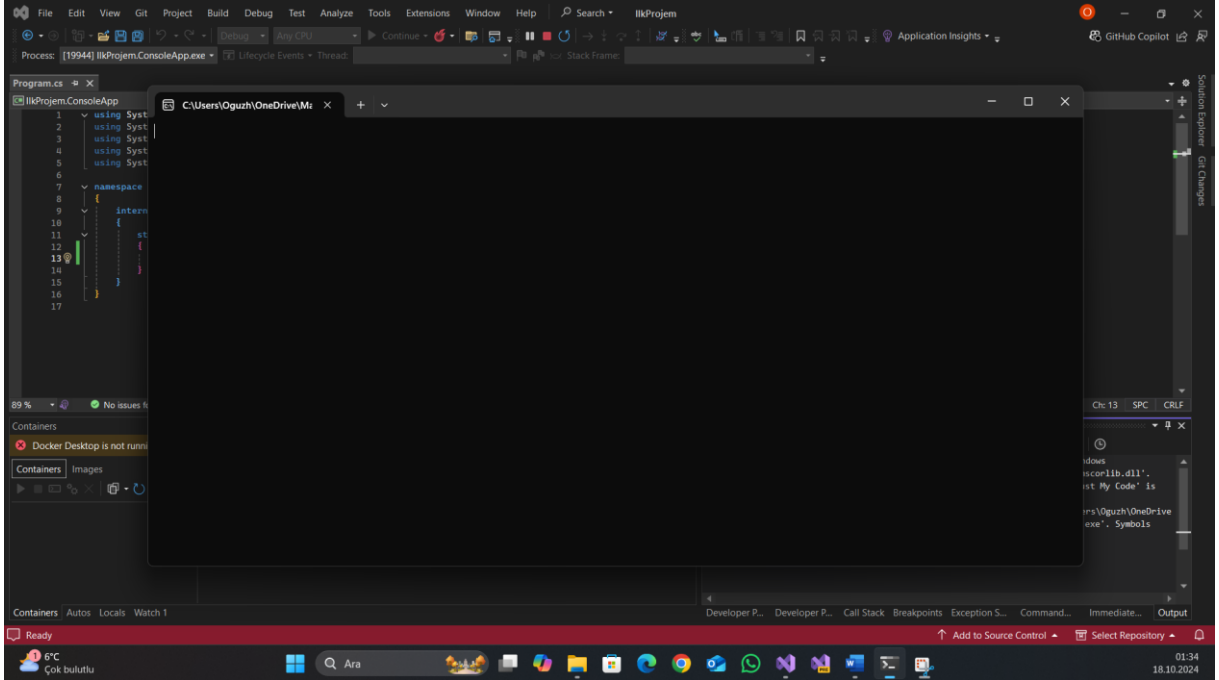
```
Console.Read();
```

Bu kod bloęu sayesinde konsol ekranımızda açık bir şekilde bekleyecektir. Aksi takdirde program çalıştırılıp çalışması bittięi için anında kapatılacaktır. Bu kod bloęu sayesinde program bizden bir girdi beklemektedir. İlk girdinizi yaptıęınız an program yine kapatılacaktır.



Şekil 8 İlk Projeyi Çalıştırma

Şekil 8'de ki gibi projenizi hazırladıktan sonra kırmızı dikdörtgen içerisine alınmış alanda bulunan tuşa (Start) tıklayınız.



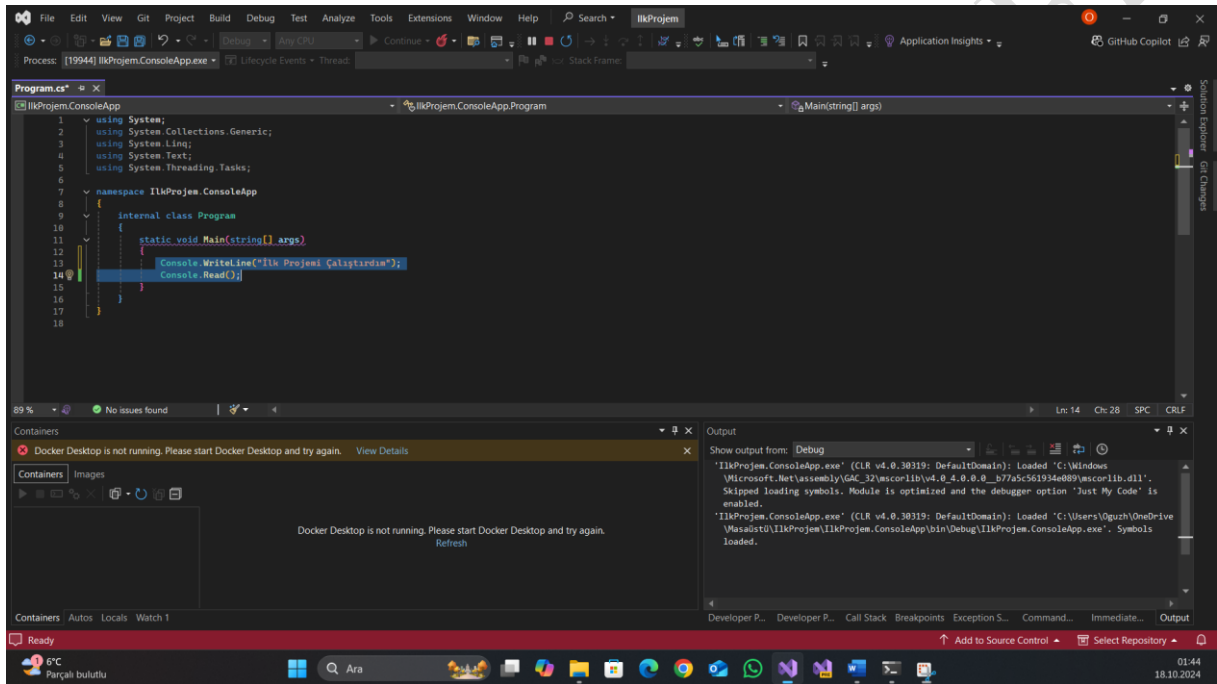
Şekil 9 Projenin Çalıştırılması.

Start tuşuna tıkladıktan sonra konsol ekranını şekil 9'da ki gibi göreceksiniz.

TEBRİKLER 🎉 İlk Projenizi oluşturduunuz ve çalıştırdınız. Hadi şimdi çok kısa bir kod yazalım ve console bir şeyler yazdıralım.

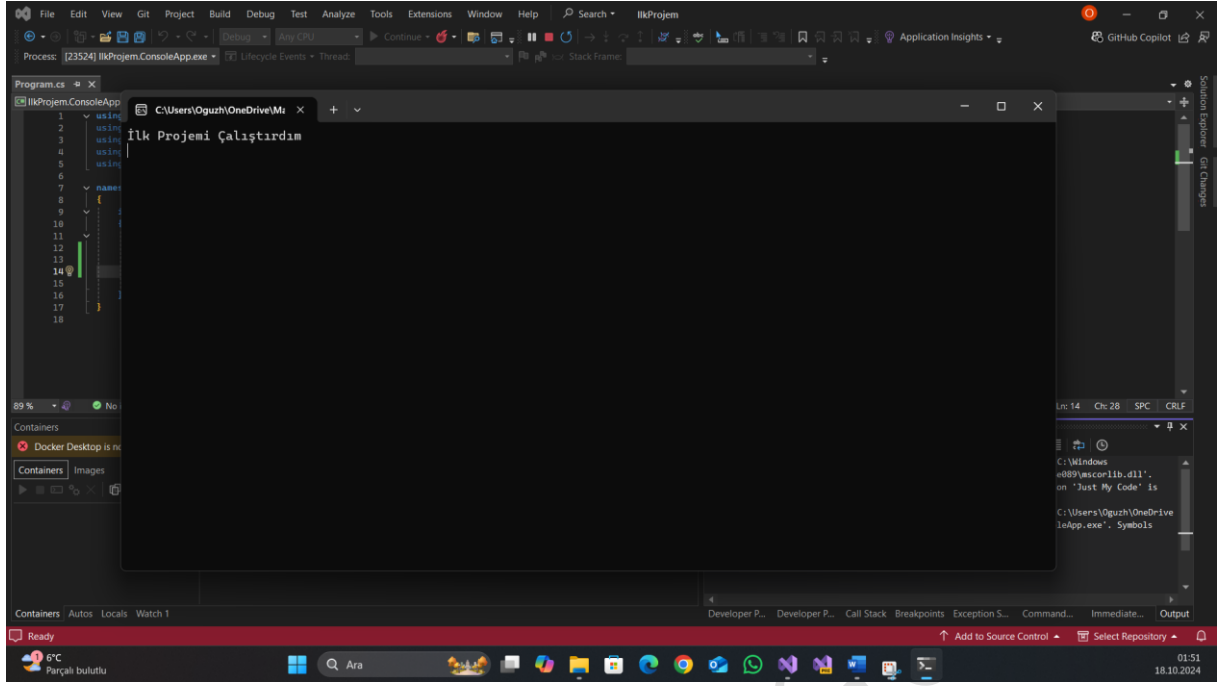
Şimdi main kod bloğunun içerisinde biraz güncelleme yapalım.

```
Console.WriteLine("İlk Projemi Çalıştırdım");  
Console.Read();
```



Şekil 10. ilk kod denemesi

Şekil 10'da ki gibi. Düzenlemeyi yaptıktan sonra projenizi tekrardan çalıştırınız. Şekil 11'de ki görüntüye ulaşacaksınız.



Şekil 11 İlk Kodu Çalıştırma

Tebrikler. İlk kod denemeniz başarılı.

Eğer buraya kadar geldiyseniz artık ilk adımı atmış oldunuz demektir. Yani en zor adımı. Bundan sonrası azim ve sıkı çalışma ile çorap söküşü gibi gelecektir. İyi çalışmalar.

## ÖDEVLER

MADDE	TAMAMLANDI	TAMAMLANMADI
Projede ki mesajı değiştirerek yeniden çalıştır. Yazdığın mesajı konsol'da gör		
Ekrana alt alta mesajlar yazdır. (Belki şimdi erken ancak ip ucu tamda gözünün önünde. Dikkatli bak!)		
Solution Explorer penceresini bi kapa aç (Belki düzelir :D, ip ucu "Solution Explorer" başlığının altında)		
Visual Studio'yu kapat ve projeni tekrardan aç. (ip ucu "Proje Oluşturma" başlığının altında)		
Hata ile karşılaşırsan yapay zeka'ya sor. Çözüme ulaştır. Eğer hata almazsan kodlarını yapay zekaya göster. (Kopyala -		

yapıştır) Yapay zekaya bu kod bloğunun ne yaptığını sor.		
--	--	--

## TEST

(\*) ile işaretlenmiş sorularda kopya çekmek serbesttir. Doğrudan araştırabilir ya da yapay zekaya sorabilirsiniz. Kopyala yapıştır yasak!

**1 – Solution nedir? (20p)**

- A)** Bir proje içindeki tüm sınıfları ve metotları organize eden dosya.
- B)** Birden fazla projeyi ve bu projelere ait dosyaları organize eden kapsayıcı.
- C)** Yalnızca Visual Studio'da kod yazmak için kullanılan bir metin dosyası.
- D)** Projelerdeki hataları ve sorunları çözmek için kullanılan bir hata ayıklama aracı

**2 \* - Aşağıdakilerden hangisi Program.cs dosyasındaki Main metodunun işlevini en iyi tanımlar?(40p)**

- A)** Main metodu, Visual Studio'daki tüm projeleri derleyen ve çalıştıran ana dosyadır.
- B)** Main metodu, C# dilinde bir uygulamanın başlangıç noktasını temsil eden ve uygulamanın çalıştığı ilk metottur.
- C)** Main metodu, uygulama boyunca kullanılan tüm sınıfları ve metodları organize eden yardımcı bir yapıdır.
- D)** Main metodu, yalnızca web projelerinde kullanılan ve tarayıcıda uygulamayı başlatan bir metottur.

**3 \* - Aşağıdakilerden hangisi Visual Studio'da ; expected hatasının anlamını en iyi şekilde açıklar? (40p)**

- A)** Kodunuzda bir noktalı virgül (;) eksik olduğu için bu hata oluşur. Bu, genellikle bir satırın sonuna noktalı virgül eklenmesi gerektiğini belirtir.
- B)** Kodunuzda bir değişken tipi eksik olduğu için bu hata oluşur. Bir değişkenin tipi belirtilmediğinde Visual Studio bu hatayı verir.
- C)** Kodunuzda gereksiz bir noktalı virgül (;) olduğu için bu hata oluşur. Bir satırda fazla noktalı virgül varsa bu hata oluşur.
- D)** Kodunuzda bir parantez (()) eksik olduğu için bu hata oluşur. Bu, genellikle metot çağrılarında parantez eksikliği anlamına gelir.

## Cevap Anahtarı

1 – B

2 – B

3 – A

Eğer bu test'ten 60 ve ya üzeri bir puan yapamadıysanız lütfen dersi konuyu tekrar ediniz.

1. Soruya yanlış cevap verdiyseniz "Solution Explorer" başlığını inceleyiniz.
2. Soruya yanlış cevap veriyorsanız "Solution Explorer" başlığını inceleyiniz.
3. Soruya yanlış cevap veriyorsanız "Output ve Error list " başlığını inceleyiniz.

Oğuzhan KARAGÜZEL