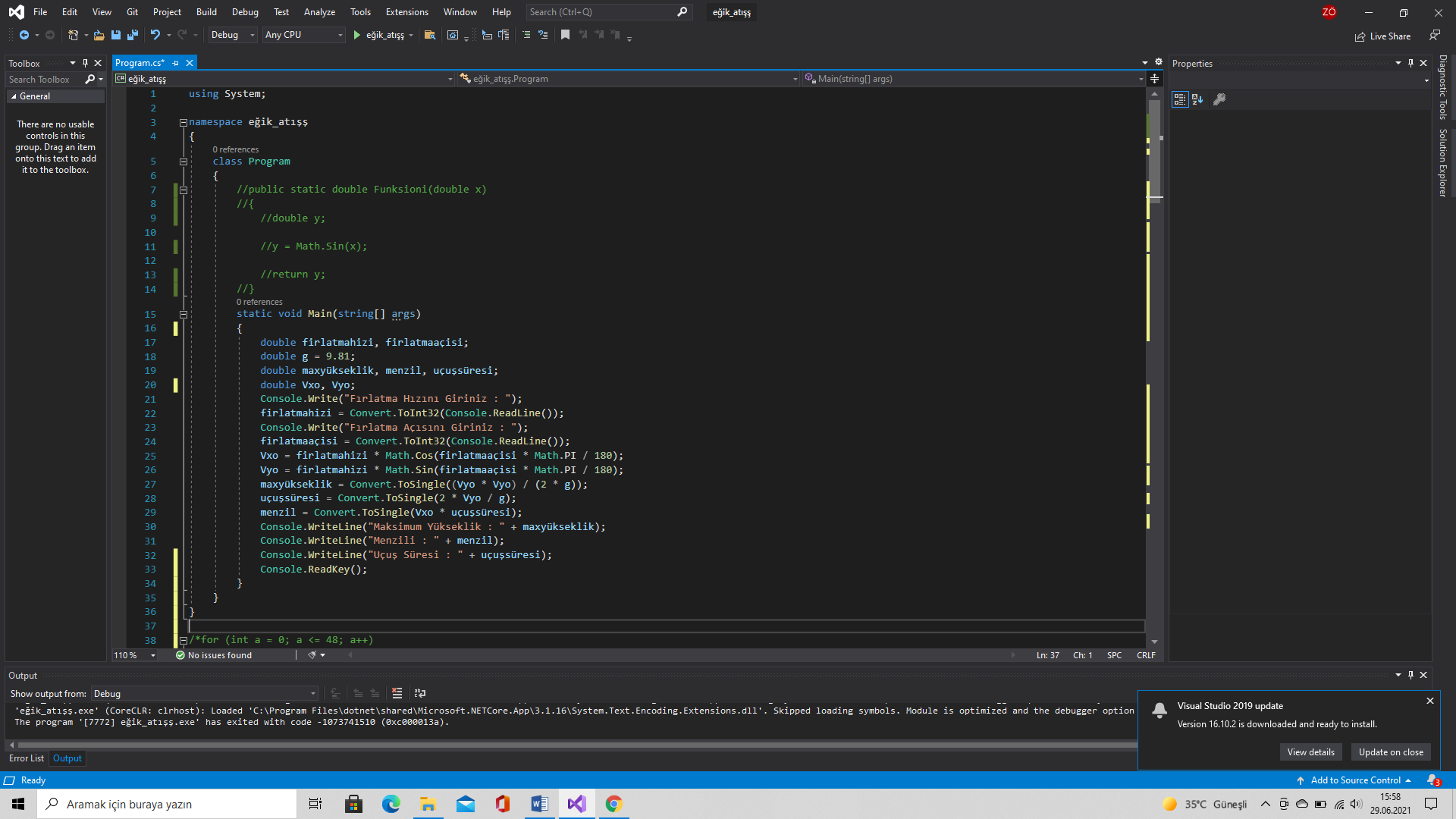
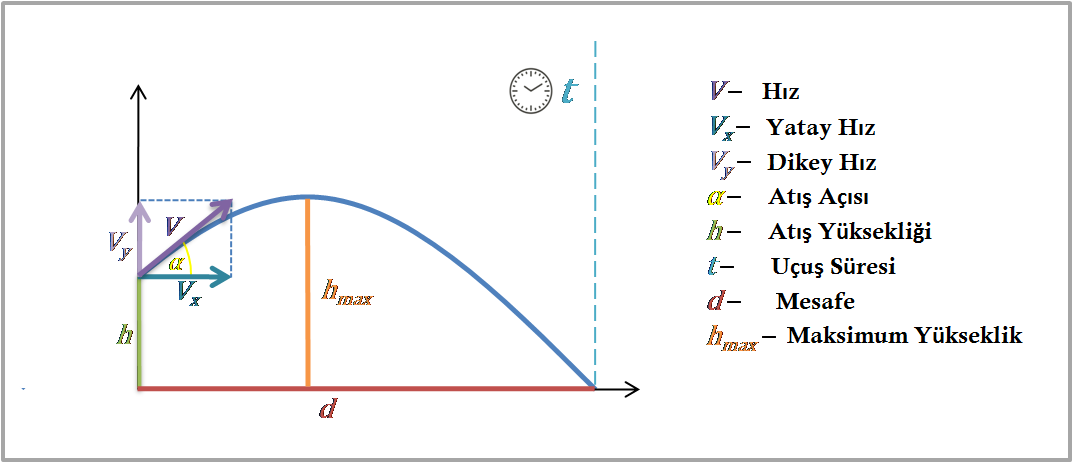
**Proje Eğik atış, menzil**

Eğik atış yerden yatay ya da düşeyde a açısı yapacak şekilde Vo ilk hızıyla atılan cisimlerin yaptığı harekettir.

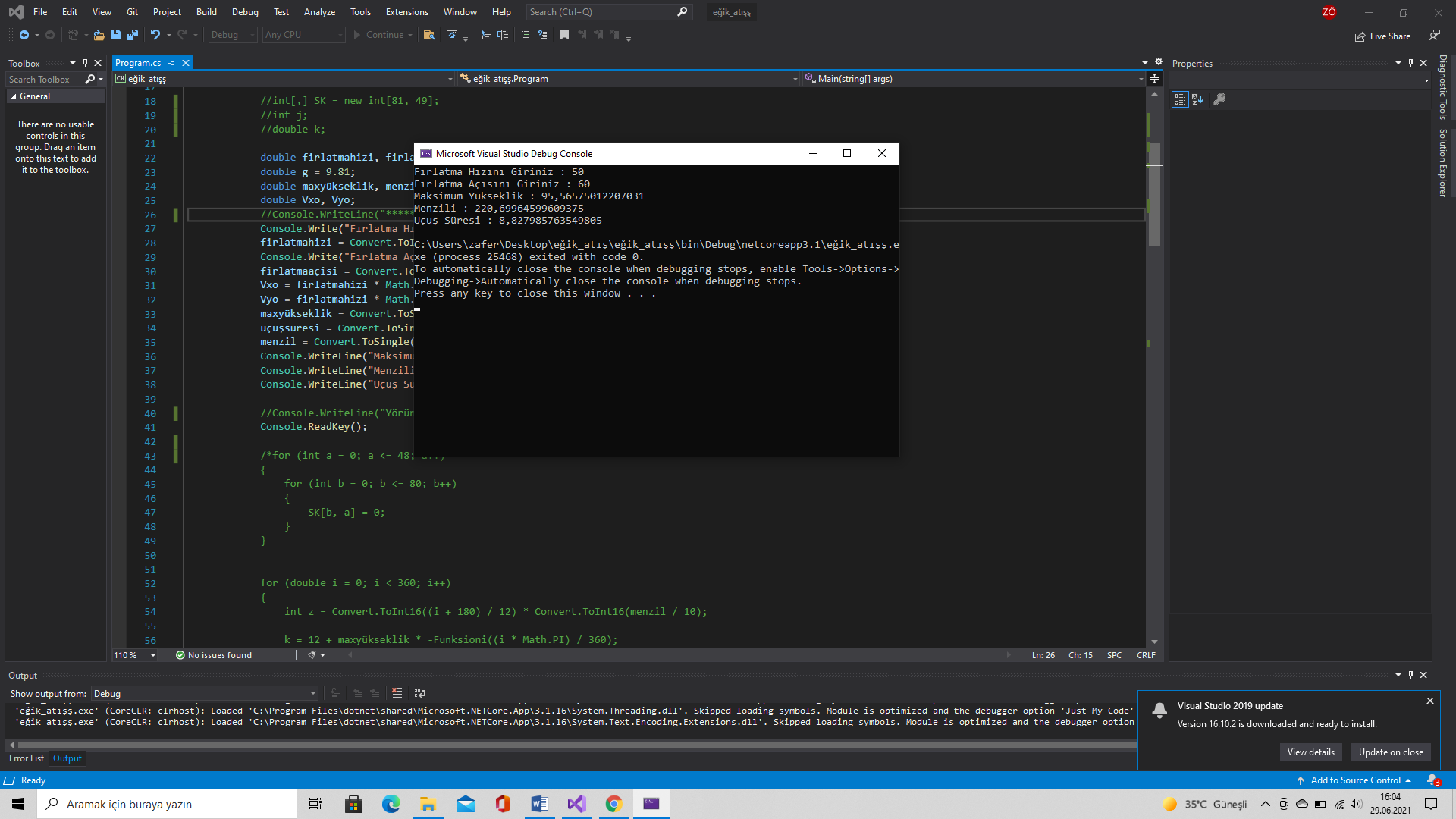
Cisim, yataydaki VoCosa ve düşeydeki VoSina hızlarıyla iki hareketi birden yapar.



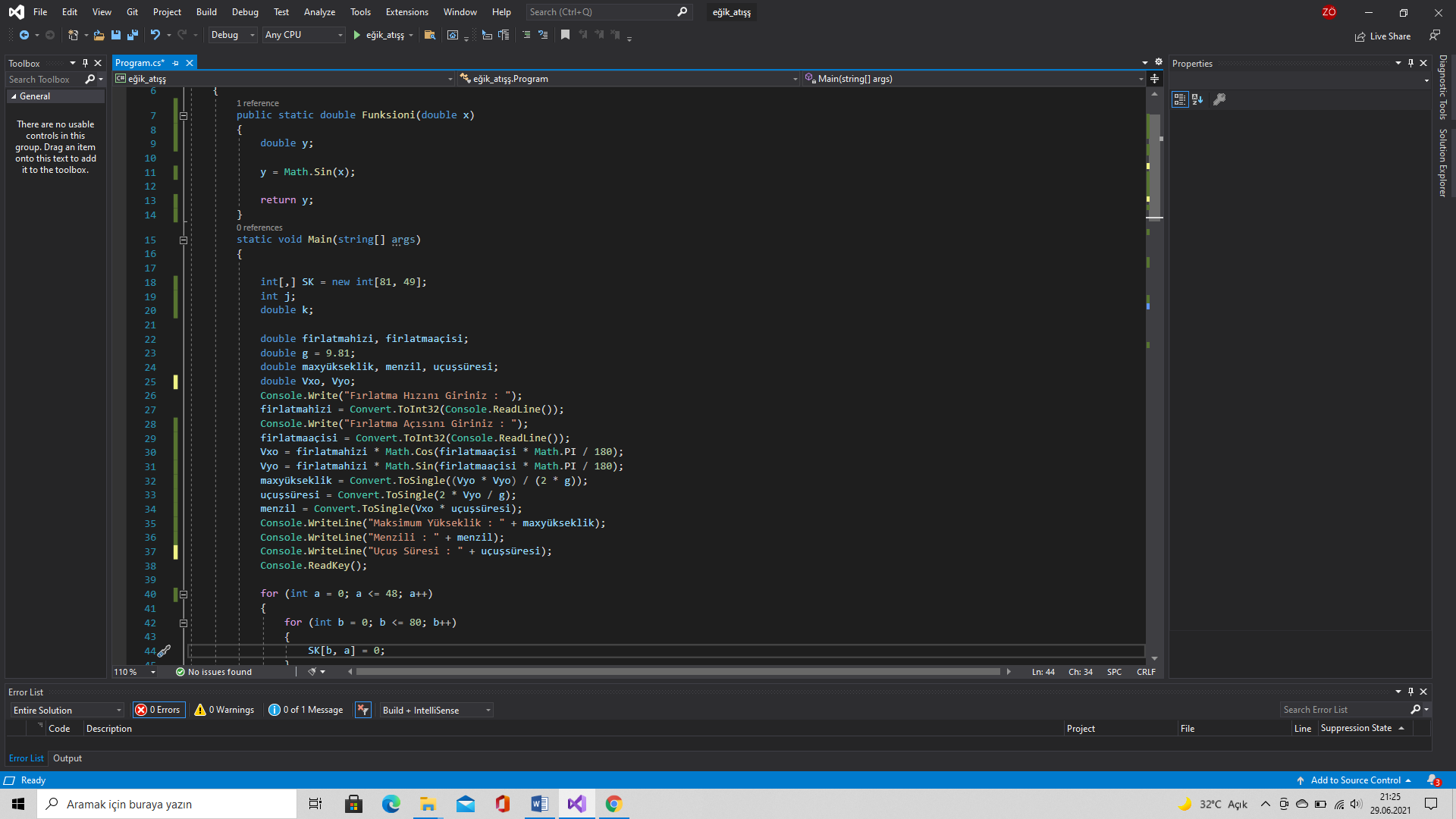
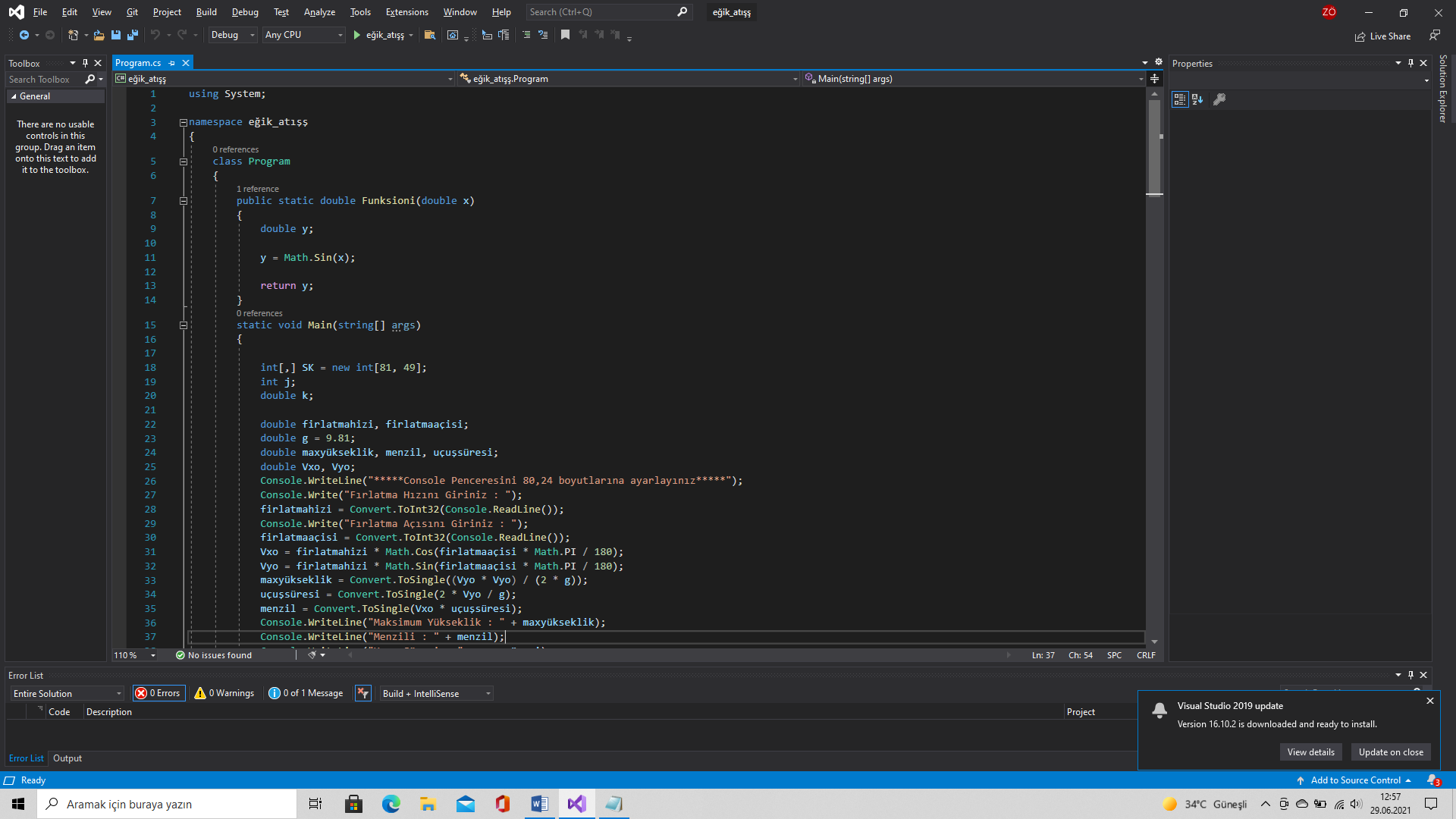
* Öncelikle main metodumuzun içine yazarak başlıyoruz.
* Fırlatma hızı, fırlatma açısı, maximum yükseklik, menzil, uçuş süresi, vxo, vyo’yu double olarak tanımlıyoruz.

g’yi (yer çekimi ivmesi. Sabit bir değerdir, 9,81) 9,81 olarak, double ile tanımlıyoruz.

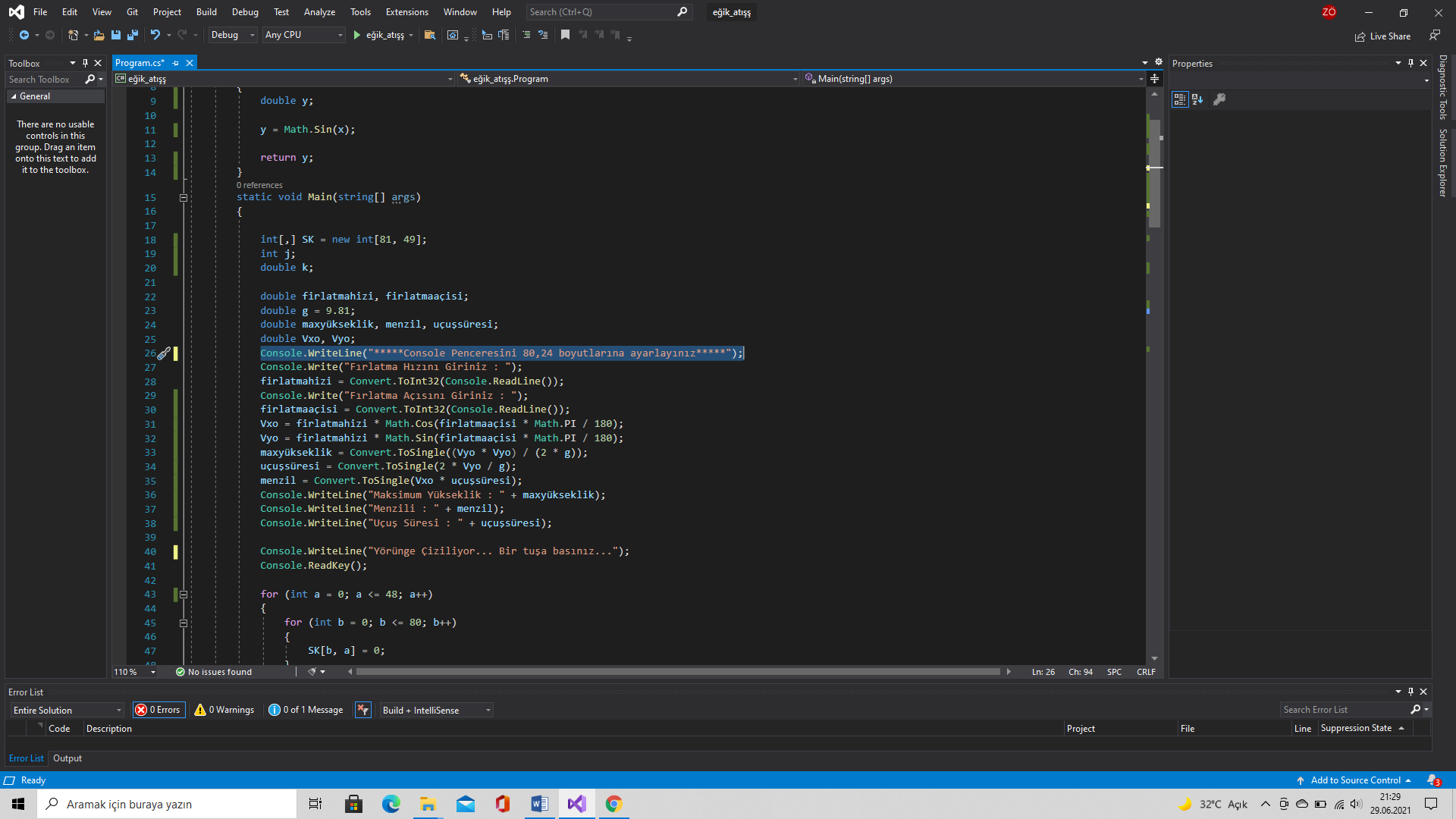
* Console.WriteLine ile konsola “Fırlatma Hızını Giriniz : ” mesajını yazdırıyoruz.
* firlatmahizi’nı çağırıyoruz. Convert.ToInt32 ile int’a dönüştürüp Console.ReadLine ile döndürüyoruz. Console.WriteLine ile konsola “Fırlatma Açısını Giriniz : ” mesajını yazdırıyoruz.
* firlatmaaçisi’nı çağırıyoruz. Convert.ToInt32 ile int’a dönüştürüp Console.ReadLine ile döndürüyoruz.
* Double veri tipini verdiğimiz vxo, vyo’yu çağırıyoruz. Vxo(yatay bileşendeki hız) için firlatmahizi’nı Math.Cos(cosinüs) metodu ile pi yöntemi çağrılır. Firlatmaaçisi ile pinin 180 ile bölümü çarpılır.
* Vyo(düşey bileşendeki hız) için aynı şekilde yöntem uyguluyoruz. maxyükseklik(maximum yükseklik) için vyo\*vyo /2\*g işlemini uyguluyoruz.(g=yer çekimi ivmesi,vyo=düşey bileşendeki hız, ‘\*’ = çarpma işlemi)
* uçuşsüresi için 2\* Vyo(düşey bileşendeki hız)/g(yer çekimi ivmesi) işlemini çağırıyoruz.
* menzil için Vxo(yatay bileşendeki hız) \*uçuş süresi işlemini çağırıyoruz.
* Console.WriteLine ile konsola maksimum yükseklik, menzil ve uçuş süresini çağırıp yazdırıyoruz. Console.ReadKey ile değeri döndürüyoruz.
* **Ekran Çıktısı:**



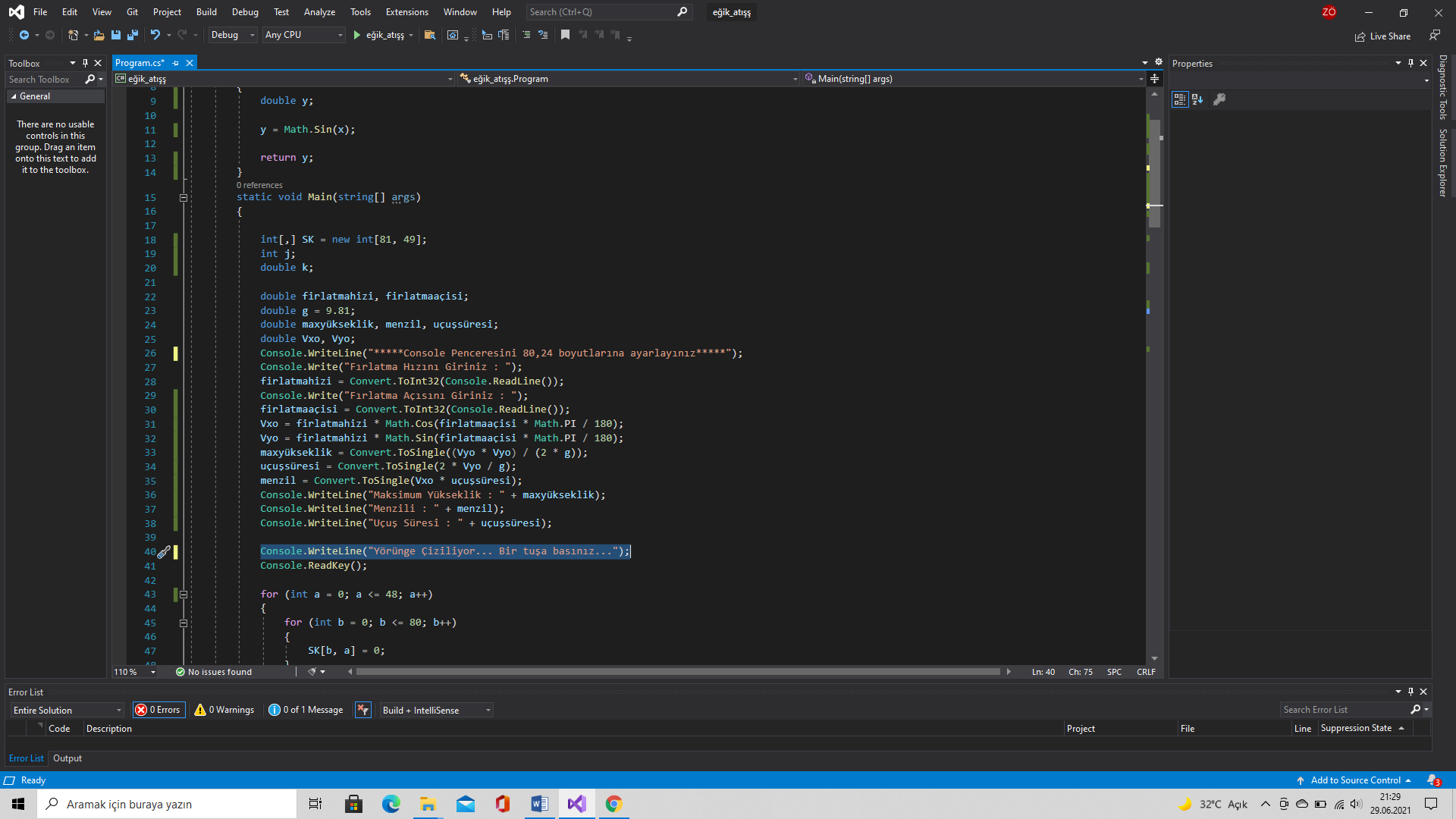
* Double ile tanımlı Funksioni metodunu oluşturuyoruz ve içinde x değişkenini double ile tanımlıyoruz.



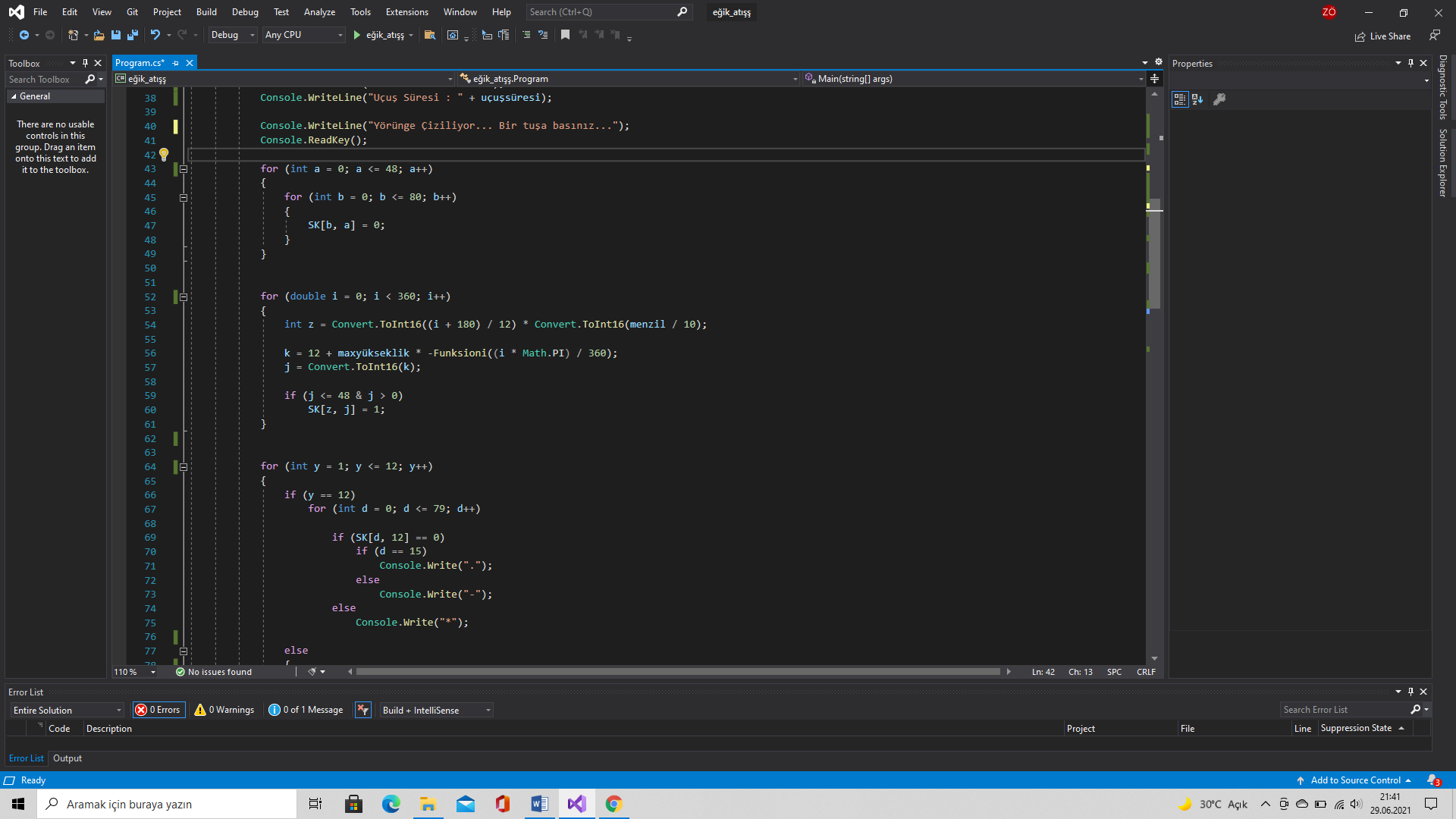
* Funksioni metodunun içinde double ile tanımlı y değişkeni oluşturuyoruz. Bunu Math.Sin metoduna atıyoruz ve y değişkenini döndürüyoruz.
* Tekrar main metoduna geliyoruz ve içine SK adında bir dizi oluşturuyoruz. SK: grafiğin verilerini 0 (boşluk) veya 1 (yıldız) olarak tutuyor. 0 ise çizim yapmıyor, 1 ise \* koyarak çizim yapıyor.
* j değişkenine int, k değişkenine double atıyoruz.



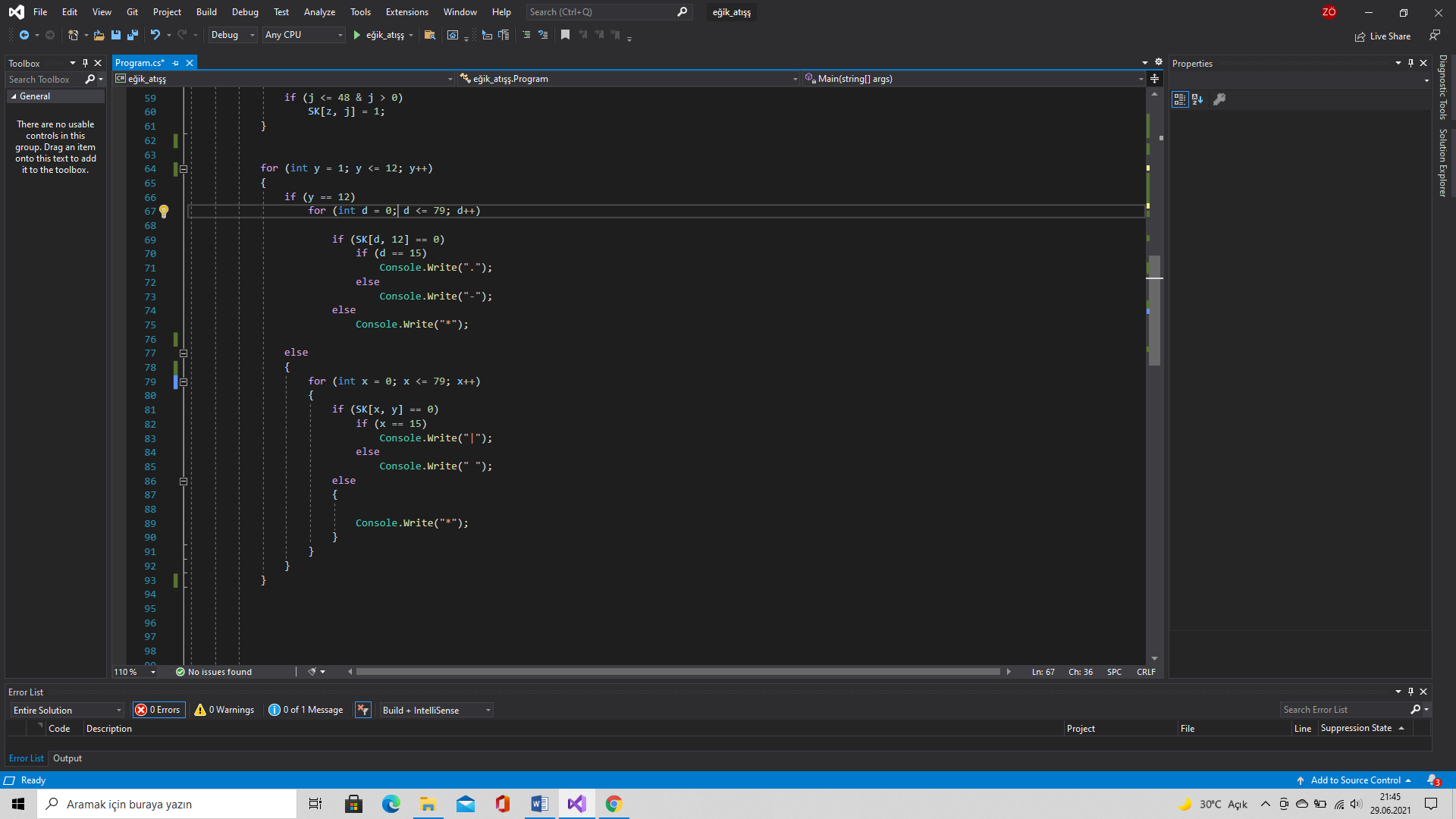
* Bu arada main metodunun içine console ekranın başına yazılacak şekilde Console.WriteLine ile “\*\*\*\*\*Console Penceresini 80,24 boyutlarına ayarlayınız\*\*\*\*\*” mesajını veriyoruz.



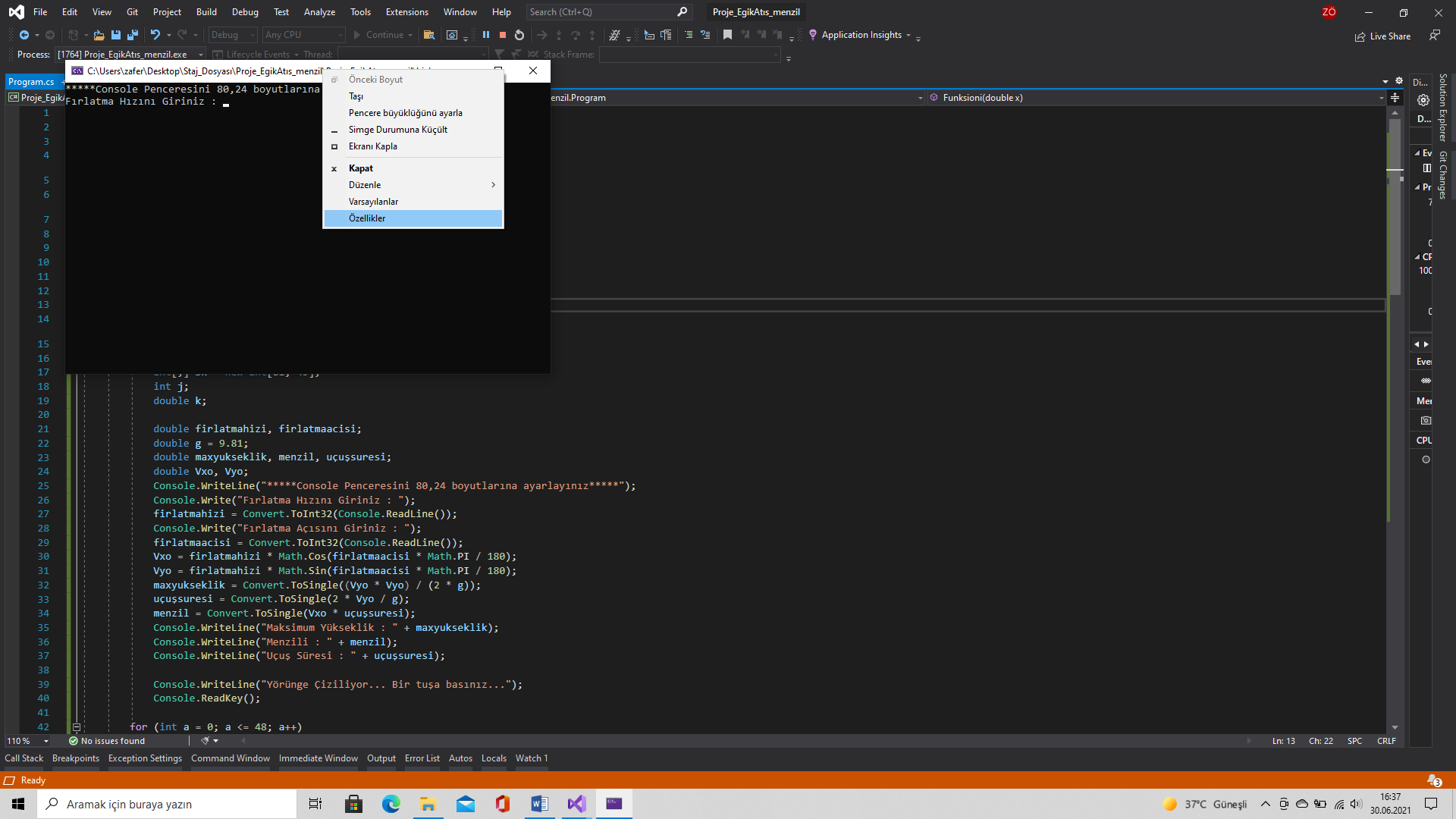
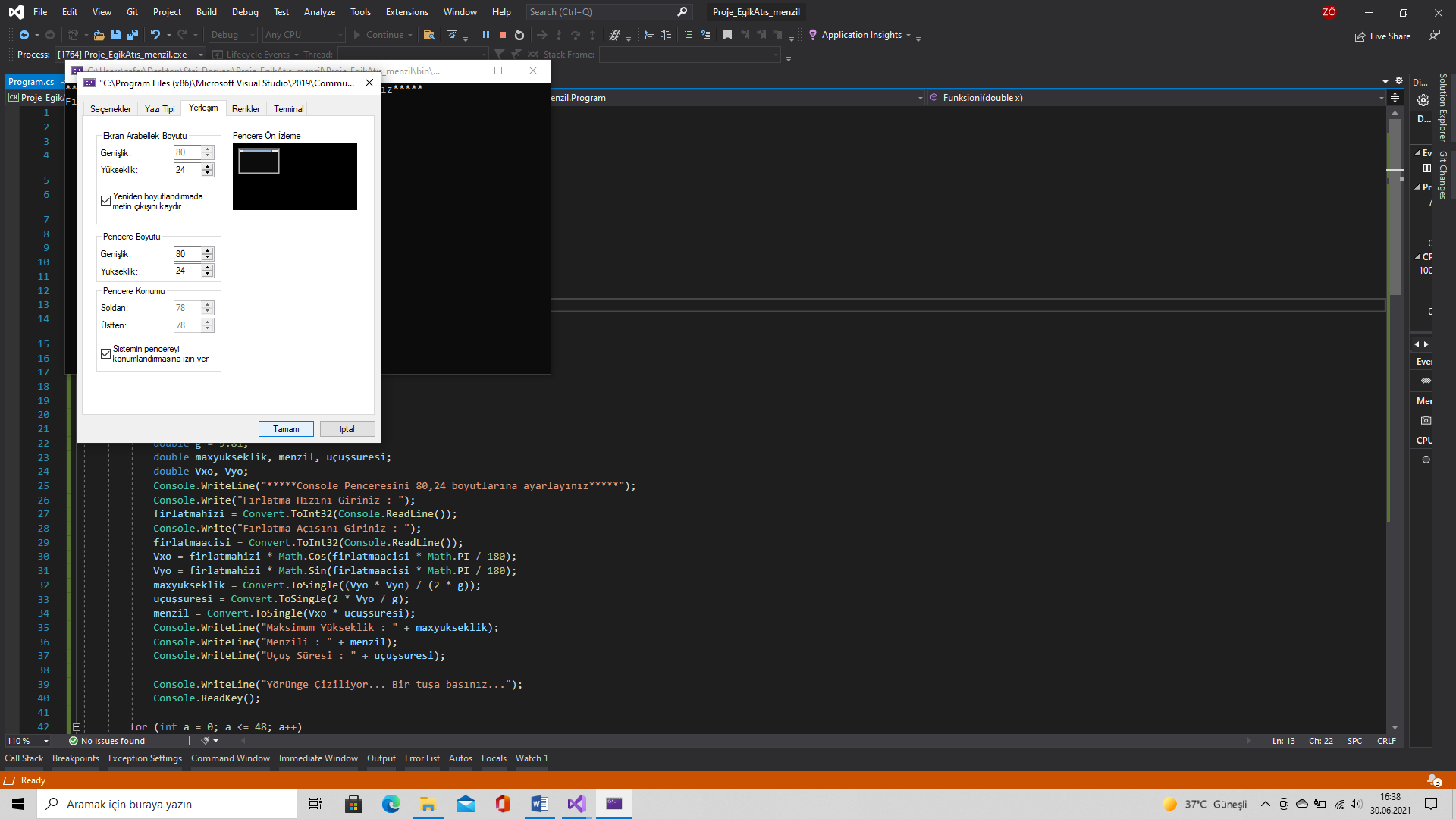
* Aynı zamanda yine Console.WriteLine ile “Yörünge Çiziliyor… Bir tuşa basınız…” mesajını konsola yazdırıyoruz.



* for döngüsü açarak 0’a eşit 48’den küçük eşit şartı ile a değişkenini int olarak tanımlıyoruz.
* İç içe for döngüsü oluşturarak 0’a eşit 80’den küçük eşit şartı b değişkenini int olarak tanımlıyoruz. Ve döngünün içine b ve a değişkenlerinin olduğu, 0’a eşit, SK adında bir Array(dizi) oluşturuyoruz.
* Yeni bir for döngüsü açarak double olarak tanımlı i değişkenini 0’a eşit ve 360’dan küçük şekilde tanımlıyoruz.
* **i döngüsü:** sayılara göre grafikte \* koyulacak değerlerinin (yani 1lerin) elde edilmesi için.
* Döngünün içine gelirsek, z adında int olarak tanımlı değişkenimizi yukarıda double olarak tanımlı (i + 180)/12) \* (menzil/10) işlemine tanımlıyoruz.(Convert.ToInt16:int değerine atamak, döndürmek için)
* Main metodumuzun başında double olarak tanımladığımız k değişkenini 12+maxyükseklik \* -Funksioni((i\*Math.PI)/360) işlemine tabii tutuyoruz.(Math.PI:pi yönetmi için. Yukarıda belirttiğimiz i değişkenini pi yöntemiyle Funksioni metodunun tersine tabii tutuyoruz,eksi haline.) j değişkenini k değişkeni ile convertliyoruz.
* if döngüsü açarak j değişkenini (0,48] aralığında tanımlıyoruz ve SK dizimizi 1 olacak şekilde çağırıyoruz.(yani yukarıda for döngüsü için if eğer anlamına geliyor, burada şart verdik)
* **z:** Grafiğin x,y noktalarının(yıldız koyulacak yerler) x değerlerini tutuyor.
* **k ve j(knın tamsayıya dönüştürülmüş hali):** Grafiğin x,y noktalarının(yıldız koyulacak yerler) y değerlerini tutuyor.
* Bir for döngüsü daha oluşturuyoruz ve içine if else döngüsünü oluşturuyoruz.(buradaki amaç yörüngeyi konsol ekranında gösterebilmemiz için yukarıda tanımladıklarımız ile şekil oluşturmak.)



* **y döngüsü :** yazım işlemi için yani grafikte uygun yerlere xy eksen çizgilerinin koyulması, boşlukların veya \* larının koyulması için.
* y değişkeni (1,12] denk olduğunda eğer y ==12 ise bir işlem, else(değilse) başka bir işlem uygulatacağız.
* Gelelim ilk işleme; iç içe if ve else ile oluşturuyoruz. Eğer SK[d,12] == 0 ise ve d==15 ise konsol ekranına “.” yazdırıyoruz. Değilse “-“ yazdırıyoruz. O da değilse “\*” yazdırıyoruz.
* Eğer en başta başka bir işlem dediğimiz ise(yani eğer y==12 değilse) yine for döngüsü kullanarak x değişkenini (0,79) tanımlıyoruz. Ancak şart koyarak.
* Eğer dizimiz==0 ise x==15 ise konsol ekranına “|” yazdır, değilse “ ”(boş) Eğer ilk if döngümüz değilse konsol ekranına “\*” yazdır.
* **Ekran Çıktısı:**
* Öncelikle açılan konsol ekranımızın boyutunu 80,24 olacak şekilde ayarlıyoruz.



* Ardından fırlatma hızı ve açısını giriyoruz. Maksimum yükseklik, menzili ve uçuş süresi hesaplanıyor.
* Yörüngenin çizilmesi için bir tuşa basın. Ve konsol ekranımıza yörüngemiz geldi.
* ÖNEMLİ NOT!!!

HIZI EN FAZLA 10, AÇISI EN FAZLA 75’E KADAR ÇALIŞIYOR, HIZ AÇIDAN DAHA KÜÇÜK OLMALI.

