

# C# Console Komisyon Hesaplama

```
E:\Udemy\SizdenGelenler\SizdenGelenler05\SizdenGelenler05\bin\Debug\SizdenGelenler05.exe
1. Ürün Fiyatını Giriniz:
45
2. Ürün Fiyatını Giriniz:
65
3. Ürün Fiyatını Giriniz:
300
4. Ürün Fiyatını Giriniz:
96
5. Ürün Fiyatını Giriniz:
434
Alınan Toplam Komisyon: 23,73
Sırayla Ürün Komisyonları
1. Ürün Komisyonu: 1,35
2. Ürün Komisyonu: 1,63
3. Ürün Komisyonu: 7,50
4. Ürün Komisyonu: 2,40
5. Ürün Komisyonu: 10,85
```

Bir komisyoncu sattığı mallardan fiyatı 50 TL kadar olanlardan %3, daha fazla olanlardan ise %2 komisyon almaktadır. Klavyeden girilen teker teker girilen 5 malın komisyonlarını bulup ekrana yazdıran ve en sonunda da toplam komisyonu ekrana yazdıran programını yazınız.

## [Komisyon hesaplama](#)

<https://github.com/saitorhan/SizdenGelenler05>

[0](#) forks.

[0](#) stars.

[0](#) open issues.

Recent commits:

- [Update Program.cs](#), GitHub
- [Update README.md](#), GitHub
- [Update Program.cs](#), GitHub
- [Create README.md](#), GitHub
- [Komisyon hesaplama](#), Sait ORHAN

# (C# Console, Diziler) Meteoroloji Sıcaklık Analizi

```
E:\Udemy\SizdenGelenler\SizdenGelenler04\SizdenGelenler04\bin\Debug\Sizd
24 gününün sıcaklık değerini giriniz
1
25 gününün sıcaklık değerini giriniz
4
26 gününün sıcaklık değerini giriniz
6
27 gününün sıcaklık değerini giriniz
Girilen sıcaklık değeri doğru formatta değil
27 gününün sıcaklık değerini giriniz
7
28 gününün sıcaklık değerini giriniz
0
Şubat Ayına ait
Ortalama Sıcaklık: 2,57
En Düşük Sıcaklık 17. Günde: -10,00
En Yüksek Sıcaklık 7. Günde: 46,00
```

Meteoroloji merkezi için bir program tasarlanması istenilmiştir. Programın çalışma şekli ise şöyle olmalıdır:

a. İlk önce hangi ay için sıcaklık bilgisi girileceği kullanıcıya sorulacaktır.

b. Girilen ay bilgisine uygun olarak o ayda kaç tane gün var ise kullanıcıdan gün sayısı kadar sıcaklık bilgisi girilmesi istenilecektir (şubat ayı için gün sayısını 28 alınız).

c. Sıcaklık veri girişi bittikten sonra o ayın sıcaklık

ortalaması ve en düşük sıcaklık bilgisi ekrana yazdırılacaktır. Bu işlemden sonra program sonlanacaktır. Örnek Çıktı: Şubat Ayına ait Ortalama Sıcaklık=15,6 derecedir ve En düşük sıcaklık 6.Gün=10, 1 derecedir.

### Sıcaklık değeri analizi

<https://github.com/saitorhan/SizdenGelenler04>

[0](#) forks.

[0](#) stars.

[0](#) open issues.

Recent commits:

- [Update README.md](#), GitHub
- [Create README.md](#), GitHub
- [init](#), Sait ORHAN
- [Add project files.](#), Sait ORHAN
- [Add .gitignore and .gitattributes.](#), Sait ORHAN

---

# C# Console Toplama İşlemi Oyunu

```
file:///E:/Udemy/SizdenGelenler/SizdenGelenler03/SizdenGelenler03/bin/Debug/SizdenGe
Aşağıdaki işlem sonuçlarını giriniz:
36 + 33 = ?99
Üzgünüm Bilemediniz
Tekrar oynamak istiyor musunuz? (e / E)
e
81 + 11 = ?92
Tebrikler Bildiniz
Tekrar oynamak istiyor musunuz? (e / E)
e
74 + 79 = ?153
Tebrikler Bildiniz
Tekrar oynamak istiyor musunuz? (e / E)
h
Puan: 8
Doğru Cevap: 2
Yanlış Cevap: 1
```

Toplama işlemini öğretmeye çalışan bir oyun programı yazılacaktır. Oyun başladığı zaman rastgele 2 tane 1-100 arasında sayı tutulacak, tutulan sayılar ekrana gösterilecek ve kullanıcıya bu sayıların toplamı nedir diye sorulacaktır. Eğer kullanıcı doğru cevap verirse “Tebrikler Bildiniz” Değil ise “Üzgünüm Bilemediniz” diye mesaj verecektir. Her cevaptan sonra “Tekrar Oynamak istiyormusunuz(e/E)?” şeklinde bir soru sorulacak ve eğer kullanıcı “e” veya “E” ile karşılık verirse oyun tekrar başlayacaktır. Kullanıcının puanı her doğru cevap için 5 puan artacak, her yanlış cevap için ise 2 puan azalacaktır. Oyun sonlandığında kullanıcının verdiği doğru cevap sayısı, yanlış cevap sayısı ve puanı ekranda listelenmelidir. Bu işlemleri yapan programın C# kodlarını yazınız.

## [Toplama oyunu](#)

<https://github.com/saitorhan/SizdenGelenler03>

[0](#) forks.

[0](#) stars.

[0](#) open issues.

Recent commits:

- [Update README.md](#), GitHub

- [Update README.md](#), GitHub
- [Create README.md](#), GitHub
- [init](#), Sait ORHAN
- [Add project files.](#), Sait ORHAN

# C# Sayıyı Basamaklarına Ayırma

```
file:///E:/Udemy/SizdenGelenler/SizdenGelenler02/SizdenGelenler02/Con
Basamaklarına ayrılacak sayıyı giriniz:
765467
Girilen Sayının Basamak sayısı: 6
Sayının Basamakları:
7 tane Yüz Binlik
6 tane On Binlik
5 tane Binlik
4 tane Yüzlük
6 tane Onluk
7 tane Birlik
```

Klavyeden girilen herhangi bir sayıyı basamaklarına ayırıp her bir basamak değerini ve sayının kaç basamaklı olduğunu ekrana yazan C# programını yazın? Not: Hazır string ya da char fonksiyonlar kullanmayın. Girilen sayının basamak sayısı belli değildir. Her hangi bir sayı girildiğinde program doğru bir şekilde çalışmalıdır. Örneğin 123 girilmişse programın ekran çıktısı: 1 tane 100'lük, 2 tane 10'lük, 3 tane 1'lik; sayı 3 hanelidir şeklinde olmalıdır.

[Sayıyı basamaklarına ayırma](#)

<https://github.com/saitorhan/SizdenGelenler02>

0 forks.

0 stars.

0 open issues.

Recent commits:

- [Update README.md](#), GitHub
- [edit](#), Sait ORHAN
- [edit](#), Sait ORHAN
- [Update README.md](#), GitHub
- [init](#), Sait ORHAN

---

# C# Rastgele Sayı Dizisi Oluşturup Ortalamaya En Yakın X Adet Sayıyı Ekranı Gösterme

```
file:///E:/Udemy/SizdenGelenler/SizdenGelenler/ConsoleApplication1/bin/Debug/Co
Başlangıç değerini giriniz
10
Bitiş değerini giriniz
100
İstenen Sayı adetini giriniz
40
Gösterilecek Sayı adetini giriniz
7
Rastgele oluşturulan sayıların ortalaması: 56,150
Ortalamaya en yakın 7 sayı
55
59
53
61
51
63
49
```

Klavyeden girilen başlangıç ve bitiş değerleri arasında, klavyeden girilen sayı kadar rastgele tek sayı üretip bu sayılardan ortalamaya en yakın olan x adet sayıyı ekrana yazdıran C# programını yazınız?

### Rasgele sayı üretip, ortalamaya en yakın sayılar

<https://github.com/saitorhan/SizdenGelenler01>

0 forks.

0 stars.

0 open issues.

Recent commits:

- [Update README.md](#), GitHub
- [a](#), Sait ORHAN
- [edit](#), Sait ORHAN
- [Merge branch 'master' of https://github.com/saitorhan/SizdenGelenler01](#), Sait ORHAN
- [init](#), Sait ORHAN

---

## Konsol Mayın Tarlası



Programlamaya girişte yazılan en güzel uygulamalardan biri de mayın tarlasıdır. Projemizde C# ile yazılmış bir mayın tarlası yer almaktadır. Projeye aşağıdaki linkten ulaşabilirsiniz:

<https://github.com/saitorhan/MayinTarlasiConsole>

---

# Kodlamada Dikkat Edilecek Noktalar

**Döngü içerisinde yeni değişken tanımlaması yapmayın.**

Örneğin;

```
int toplam = 0;
for (int i = 0; i < 1000; i++)
{
    int sayi = i * 10;
    toplam += sayi;
}
```

döngüsü için sürekli aynı işlemi gören “sayi” değişkeni için bellekten 1000 adet ayrı hücre tahsis edilir, bu işlem de belli bir gecikmeye neden olur. Bunun yerine;

```
int toplam = 0;
int sayi;
for (int i = 0; i < 1000; i++)
{
    sayi = i * 10;
    toplam += sayi;
}
```

şekline “sayi” değişkeni döngü dışında tanımlanırsa bu işlem için tek bir bellek hücresi üzerinde işlem yapar.

**Noktalı sayı karşılaştırmalarını “==” işlemi ile yapmayın.**

Bunun nedeni karşılaştırma işlemi “==” ile yapıldığında



eşitlik şartı için bütün bitlerin test edilmesidir. Özellikle bölme işlemlerinin hassasiyetinden dolayı noktalı sayılarda bölme sonrası öngörülemeyen bazı küçük değer farklılıklarının olmasıdır. Bu işlem için; tolere edilebilecek bir hata değerini tespit edin ve karşılaştırılacak sayıların farkının bu hata değerinden küçük veya eşit olmasını deneyin. Örneğin;

```
double hata = 0.0001;
if (toplam - toplam2 <= hata)
{
    //////////////////////////////////
}
```