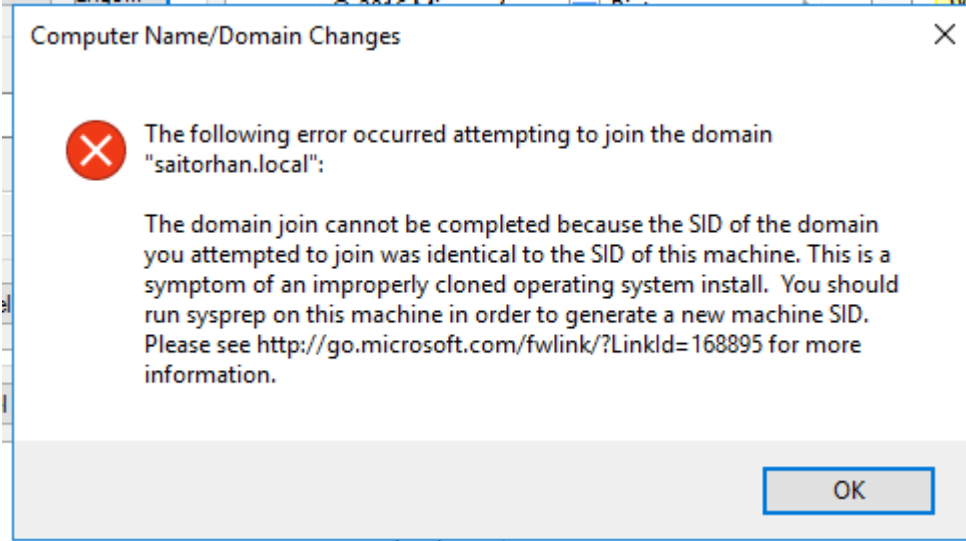
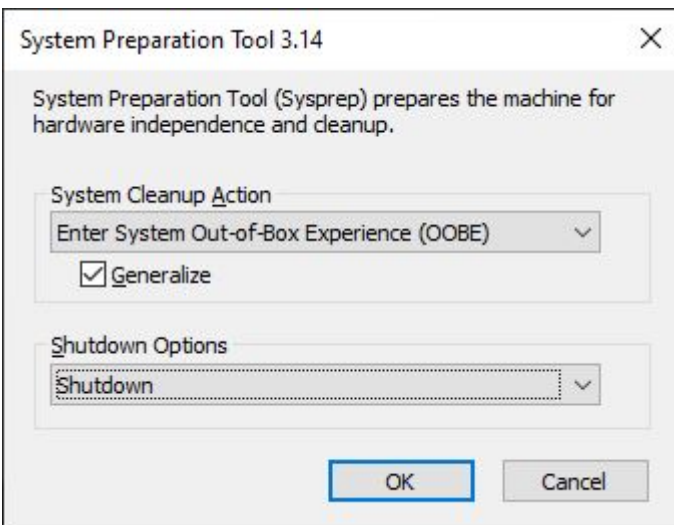


Vmware Klonlanan Bilgisayarın Domaine Alınamama Hatası



Sanal sistemde bulunan bir makine klonlandıktan sonra aynı yapıda iki tane SID numarasına sahip makine olacağından ikinci klon makinenin lisans bilgisi geçersiz olur. Aynı zamanda domaine alınmaya çalışılırken de yukarıdaki ekranda görüldüğü gibi hata verir.



sysprep Programının Kullanımı

Alınan bu hatanın çözümü için **C:\WINDOWS\System32\Sysprep\sysprep.exe** adresinde bulunan programı yönetici olarak çalıştırın. Bu işlem ile makineye yeni bir SID numarası üretir. Program çalıştırılırken dikkat edilmesi gereken nokta "Generalize" seçeneğinin seçili olması gerekiyor ve "Shutdown Options" seçeneğinin "Shutdown" olarak ayarlanmış olması gerekiyor. Bu işlemden sonra bilgisayar kapanacaktır. Tekrar açıldıktan sonra sıkıntısız şekilde domaine dahil edilebilecektir.

C# Mükemmel Sayı Testi

```
E:\Udemy\SizdenGelenler\SizdenGelenler06\SizdenGelenler06\bin\Debug\S
Test Edilecek Sayı Giriniz:
4545
4545 Sayısı Mükemmel Sayı Değildir
```

Kendisi hariç bütün pozitif çarpanları (tam bölenleri) toplamı, yine kendisine eşit olan sayılara "mükemmel sayı" denir. Örneğin $6=1+2+3$ ve $28=1+2+4+7+14$ gibi. Buna göre klavyeden girilen bir tam sayının "mükemmel sayı" olup olmadığını kontrol eden C# programını kodlarını yazınız

[Mükemmel sayı testi](#)

<https://github.com/saitorhan/SizdenGelenler06>

[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 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)

<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

PowerShell İle İşletim Sistemine Göre AD Bilgisayar Sayılarını Bulma

Count	Name
69	Windows 10 Pro
5	Windows 10 Pro for Workstations
158	Windows 7 Professional
5	Windows 8 Pro
42	Windows 8.1 Pro
1	Windows Server 2003
7	Windows Server 2008 R2 Enterprise
1	Windows Server 2008 R2 Standard
3	Windows Server 2012 Datacenter
27	Windows Server 2012 R2 Standard
6	Windows Server 2012 Standard
9	Windows XP Professional

Merhaba arkadaşlar,

[Sistem yöneticilerinin](#) özellikle lisans denetimleri öncesinde ve anti virüs yazılımları gibi bütün makinelere kurulacak sistemler öncesi saha analizinde işletim sistemlerine göre bilgisayar sayıları ihtiyaç duydukları hayati bilgilerden biri olabiliyor. AD'ye ilişkin hemen hemen her soruya cevap veren [PowerShell](#) ile bunun cevabını vermek de son derece basittir. Aşağıdaki powershell scripti ile yukarıdaki ekran görüntüsünde görüldüğü gibi bilgisayar sayıları kolaylıkla elde edilebilir.

```
# Import AD module
Import-Module ActiveDirectory

# Domain adını bulma
$DomainName = (Get-ADDomain).NetBIOSName

# Kaç gün öncesine kadar oturum açmış makineleri sorgula
$days = 30
$lastLogonDate = (Get-Date).AddDays(-$days).ToFileTime()

# AD sorgulama
$Computers = @(Get-ADComputer -Properties
Name,operatingSystem,lastLogontimeStamp -Filter
{(OperatingSystem -like "*Windows*") -AND (lastLogontimeStamp
-ge $lastLogonDate)})
foreach($Computer in $Computers)
{
    $Computer.OperatingSystem = $Computer.OperatingSystem -
replace '®' -replace '™' -replace '   ', 'Professional (Ch)' -
replace 'Professionnel', 'Professional (Fr)'
}

$Computers | Group-Object operatingSystem | Select Count,Name
| Sort Name | Out-GridView
```

Programlama Hassasiyet İpucu: Noktalı Sayı Kullanımı

```

class Program
{
    static void Main(string[] args)
    {
        double num1 = 0.1 + 0.7;
        if (num1 == 0.8)
        {
            Console.WriteLine("Correct");
        }
        else
        {
            Console.WriteLine(num1);
        }
    }
}

```

Noktalı sayı kullanılırken özellikle eşitlik karşılaştırmalarında kurguladığımız senaryo gereği iki sayının birbirine eşit olma durumu için bir tolerans aralığı seçilmelidir. Aşağıda örnek bir senaryo üzerinde ne demek istediğimizi anlatmaya çalışalım.

Örneğin demir çelik endüstrisi için bir otomasyon geliştiriyorsunuz ve birim ağırlık olarak kg kullanıyorsunuz. Böyle bir senaryoda 5000 kg ve 5000,005 kg aslında birbirine eşit kabul edilebilir bir aralıktır. Ama programlamada 5000 kg içerisinde aradaki 5 gramlık farktan ötürü eşitlik şartı sağlanmamaktadır ve eşitlik durumunda çalışması gereken kodlar çalışmayacaktır. Bunun için direk `if(sayı1 == sayı2)` şeklinde bir karşılaştırma değil de aşağıdaki gibi bir tolerans payı ile karşılaştırma yapılmalıdır.

```

const double tolerans = 0.005;
double sayı1 = 5000;
double sayı2 = 5000.005;

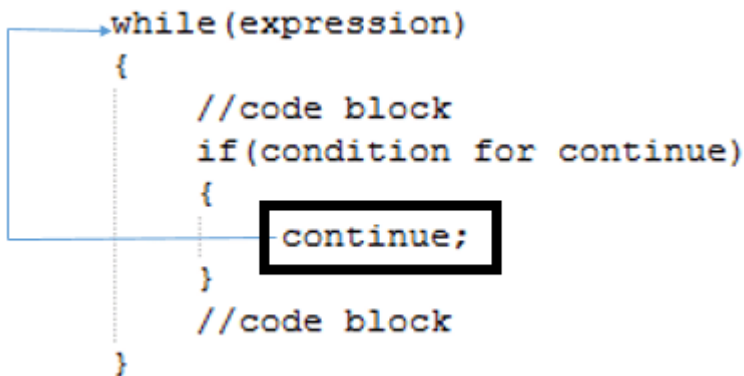
if (Math.Abs(sayı2 - sayı1) <= tolerans)
{
    // eşitlik durumu için çalışacak kodlar
}

```

Kodumuzda if şartı içerisinde aradaki toleransı (+) ve (-) durumlarının ikisinde de sağlamak için aradaki farkın mutlak değerini alıp tolerans değerinden küçük veya eşit olup olmadığının sorguluyoruz.

Programlama Performans İpucu: Döngüler (while – for – foreaach)

```
while(expression)
{
    //code block
    if(condition for continue)
    {
        continue;
    }
    //code block
}
```



Döngüler (while – for – foreaach) içerisinde atama yapılan değişkenlerin tanımlanması döngü içerisinde değil de döngü dışında tanımlanması yapılırsa performans elde edilir. Örneğin:

```
for(int i = 0; i < 10000; i++)
{
    int b = i *2;
}
```

Şeklinde b değişkeni tanımlandığında her döngüde b değişkeninin geçerlilik ömrü sona erdiğinden bellek üzerinden silinir ve b için bellek üzerinde yeni alan tahsisi yapılır, ancak bunun yerine geçerlilik ömrü bütün döngü için geçerli olacak şekilde tanımlanırsa bellek üzerinde alanın silinip yenisinin oluşturulması için gereken işlemlerden tasarruf sağlanır.

```
int b;  
for(int i = 0; i < 10000; i++)  
{  
    b = i *2;  
}
```