**Python Temel Tüm Konular**

**Python ile neler yapabilirim?**



1. **Script Yazmak**

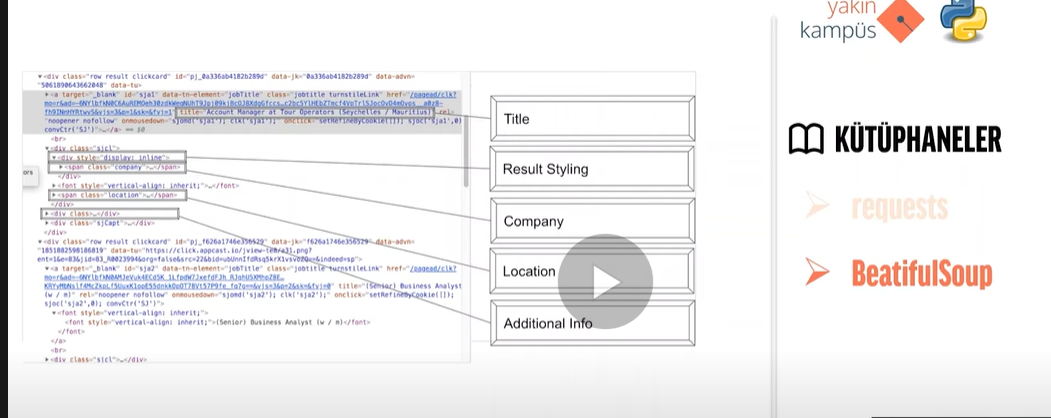
Script, bir yol çizmektir.

Otomasyon olarak, tekrar içeren durumlarda kullanılır.

Örnek üzerinden açıklamak istersek,

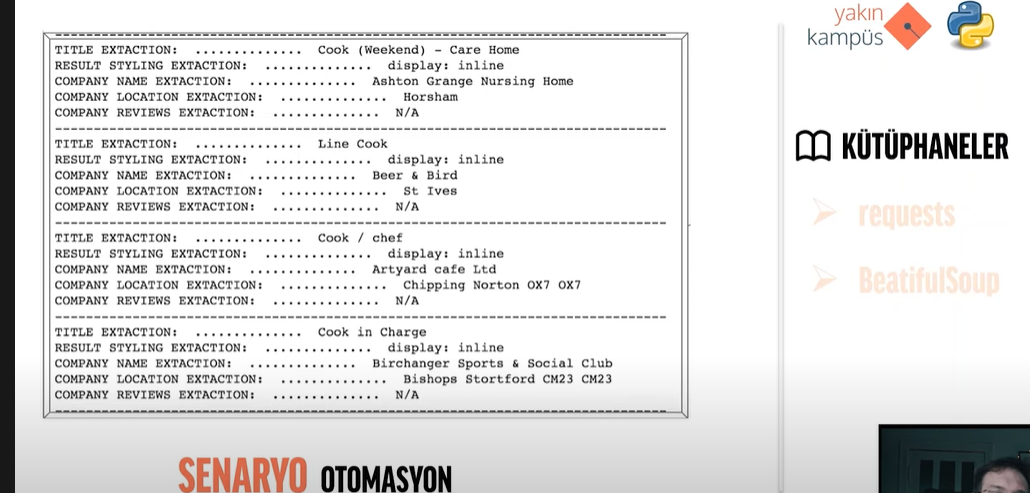
yazdığımız programın her gün aynı saatte rakip şirketin web sitesine girerek sitedeki güncel haber ve bilgileri toplaması, bunları veritabanına kaydetmesi bir script örneğidir. (Veri Kazıma)

Örnek Görsel (HTML Kodları)



Rakip şirketin web sitesinin URL, Title, kaynaklar gibi adreslerini kodumuza işleyerek ve kural, şartlar girerek (Her gün saat 09.00’da bu işlemleri gerçekleştir) veri kazımaya script denir.

Örnek Görsel2



Örnek Görsel3



Route (Güzergah, yol, rota) anlamında kullanılıyor.

Güzergahlar belirlendikten sonra ve ‘hello’ tetiklendikten sonra (Örnegin sabah saat 09.00) belirlenen işlemler gerçekleştiriliyor.

(Dinleme, okuma, veri kazıma)

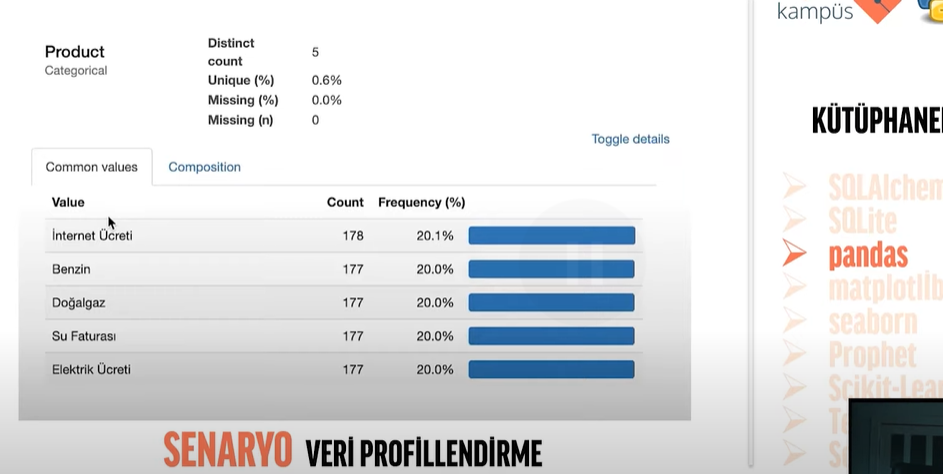
1. **Data Science (Veri Bilimi / Veritabanı)**

Veri bilimi, yapılandırılmış ve yapılandırılmamış verilerden bilgi ve öngörü elde etmek için bilimsel yöntemleri, süreçleri, algoritmaları ve sistemleri kullanan çok disiplinli bir alandır. Veri bilimi veri madenciliği ve büyük verilerle ilişkilidir

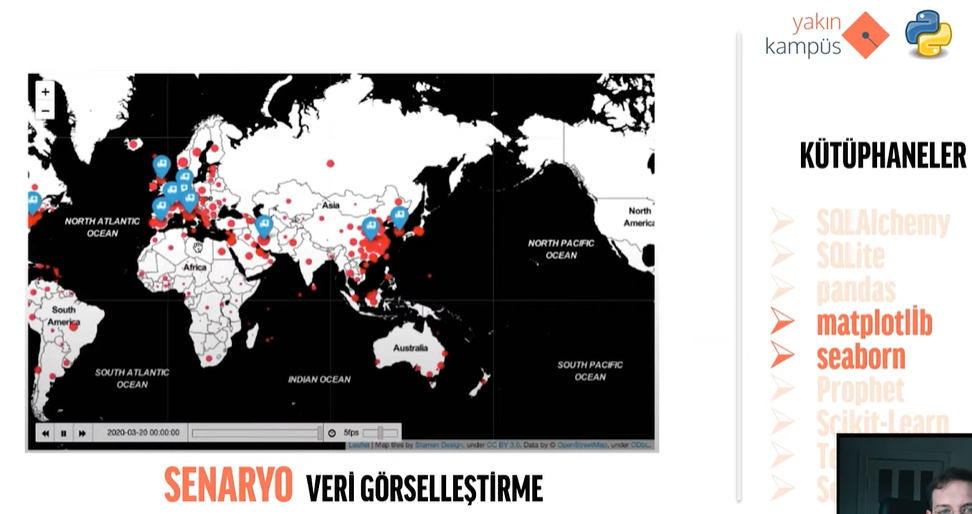
1. **Veri sorgulama ve Veri Yapılandırma**



1. **Veri Profillendirme**



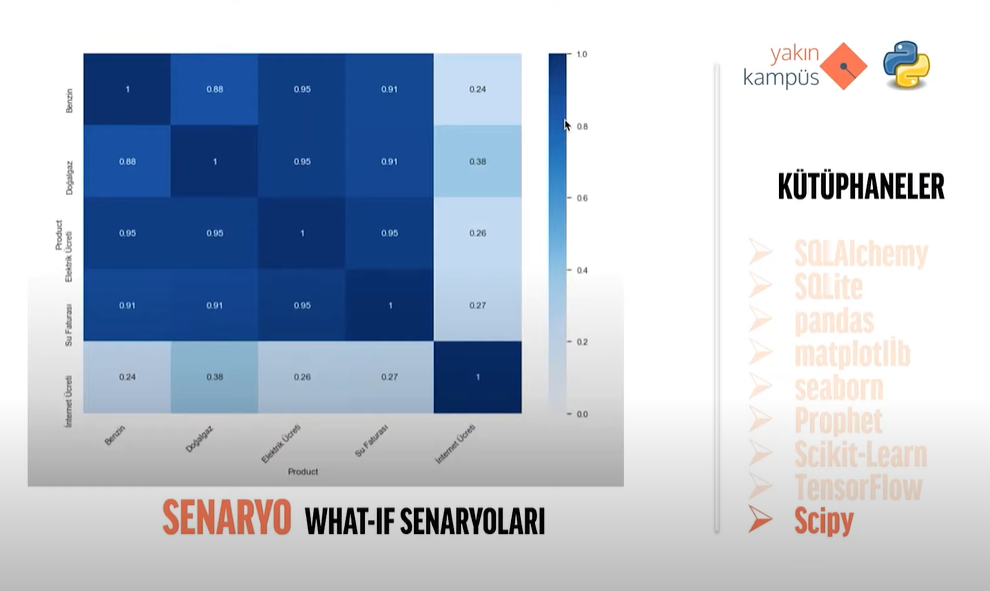
1. **Veri Görselleştirme**



1. **Veri Tahminleri**



1. **Veri İlişkileri**



1. **Machine Learning (Makine Kararı Almak)**

Makine öğrenimi veya makine öğrenmesi, bilgisayarların algılayıcı verisi ya da veritabanları gibi veri türlerine bağlı olarak öğrenmeye olanak sağlayan algoritmaların tasarım ve geliştirme süreçlerini konu edinen bir bilim dalıdır.

**Örnek bir makine öğreniminin süreçleri**

1. Gerekli kütüphanelerin import edilmesi ([Pandas](https://pandas.pydata.org/" \t "_blank), [Matplotlib](https://matplotlib.org/) , [Sklearn](http://scikit-learn.org/) kütüphanelerini)
2. Veri setinin yüklenmesi ve hazırlanması
3. Veri setinin incelenmesi
4. Veri setinin görselleştirilmesi
5. Algoritmaların uygulanması ve sonuçlarının degerlendirilmesi
6. Uygun algoritmanın seçilmesi ve tahmin yapılması

**Kaynaklar**

[**https://medium.com/bili%C5%9Fim-hareketi/python-ile-i%CC%87lk-makine-%C3%B6%C4%9Frenimi-projesi-c681250b7a48**](https://medium.com/bili%C5%9Fim-hareketi/python-ile-i%CC%87lk-makine-%C3%B6%C4%9Frenimi-projesi-c681250b7a48)

[**https://medium.com/@EbubekirBbr/makine-%C3%B6%C4%9Frenmesi-uygulamas%C4%B1-geli%C5%9Ftirme-2-python-41028d52686b**](https://medium.com/@EbubekirBbr/makine-%C3%B6%C4%9Frenmesi-uygulamas%C4%B1-geli%C5%9Ftirme-2-python-41028d52686b)

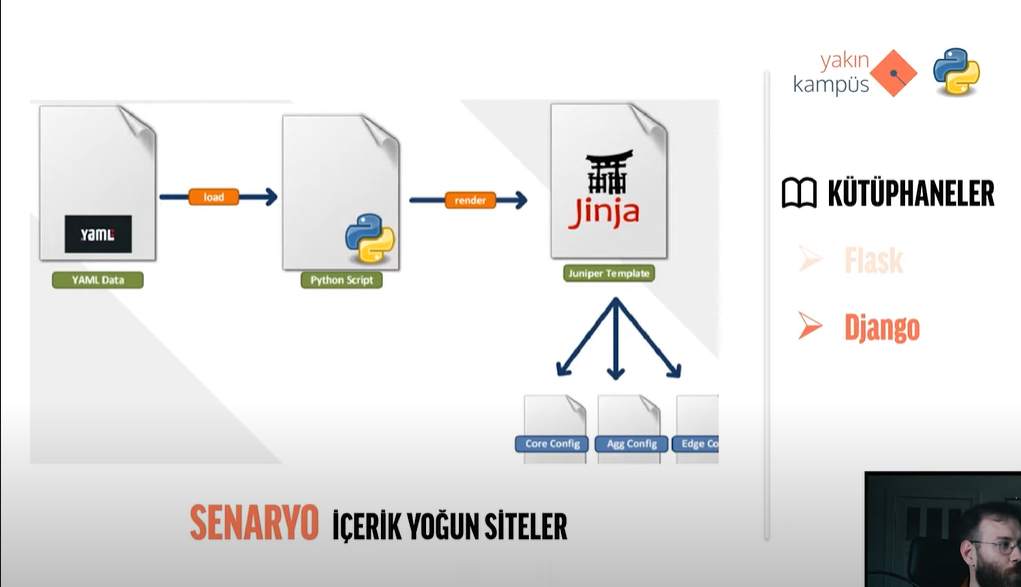
1. **Web Tasarım**

Django, Python Programlama Dili için hazırlanmış ve BSD lisansı ile lisanslanmış yüksek seviyeli bir web çatısıdır. Basit kurulumu ve kullanımı, detaylı hata raporu sayfaları ve sunduğu yeni arayüz kodlama yöntemleriyle diğer sunucu yazılımı ve çatılardan kendini ayırmaktadır.

İçeriği yoğun platformlarda Django kullanılıyor.

Youtube, Netflix, Spotify, Dropbox, İnstagram gibi platformlar Django ile tasarlanmıştır.

Web Sitesi oluşturma süreci şu şekilde gerçekleşiyor.



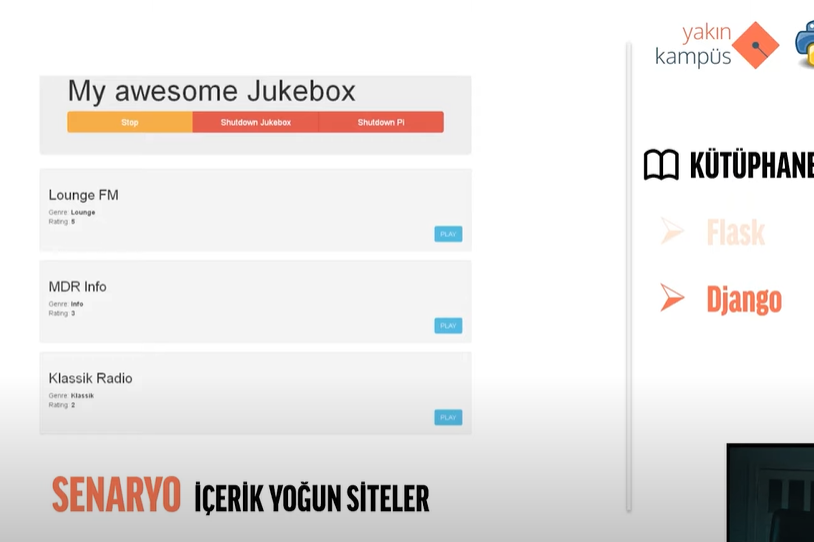
- Her internet sitesi için farklı tasarımlar vb. oluşturulmuyor.

- Gerekli datalar, malzemeler YAML ve SCRİPT ile sağlanıyor ve en nihayetinde ise Jinja ile bu verilen data vb. şeyler kullanılarak örnek bir şablona oturtuluyor.

- Jinja, Python programlama dili için oluşturulmuş bir tasarım motorudur. Örnek Şablonlar içermektedir.

-> Yukarıda bahsedilen tüm bu süreç Bootstrap olarak adlandırılıyor (Ön Yükleme)

Örnek Şablon Gösterimi



Daha detaylı bilgiler için inceleyebilirsin (Django learning series and django’s about )

<https://www.hosting.com.tr/blog/visual-studio-code-ide/>

<https://meltemdizdar.gitbooks.io/bilgisayar-kavramlari-ile-django/content/python-django-serisi-giris-1.html>

<https://www.youtube.com/watch?v=l1EQ2GfxmUg&list=PLh9ECzBB8tJMkq3vJ8fwAKNlwPA8d629n> (Merve Dizdar Youtube Series)

<https://docs.microsoft.com/tr-tr/visualstudio/python/python-django-web-application-project-template?view=vs-2022>

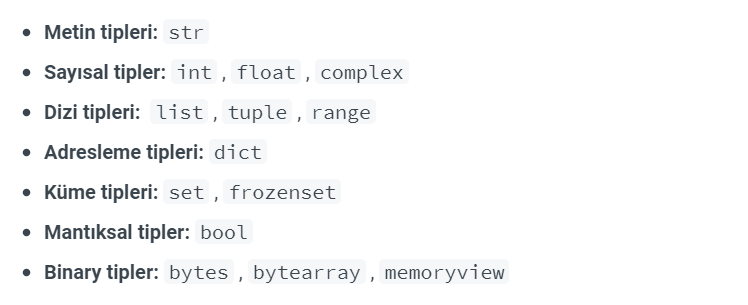
<https://zeysert.medium.com/django-i%CC%87le-web-si%CC%87tesi%CC%87-yapmak-e56b49a6b742>

https://tirendazakademi.medium.com/python-django-nedi%CC%87r-nasil-y%C3%BCkleni%CC%87r-863ef7a492da

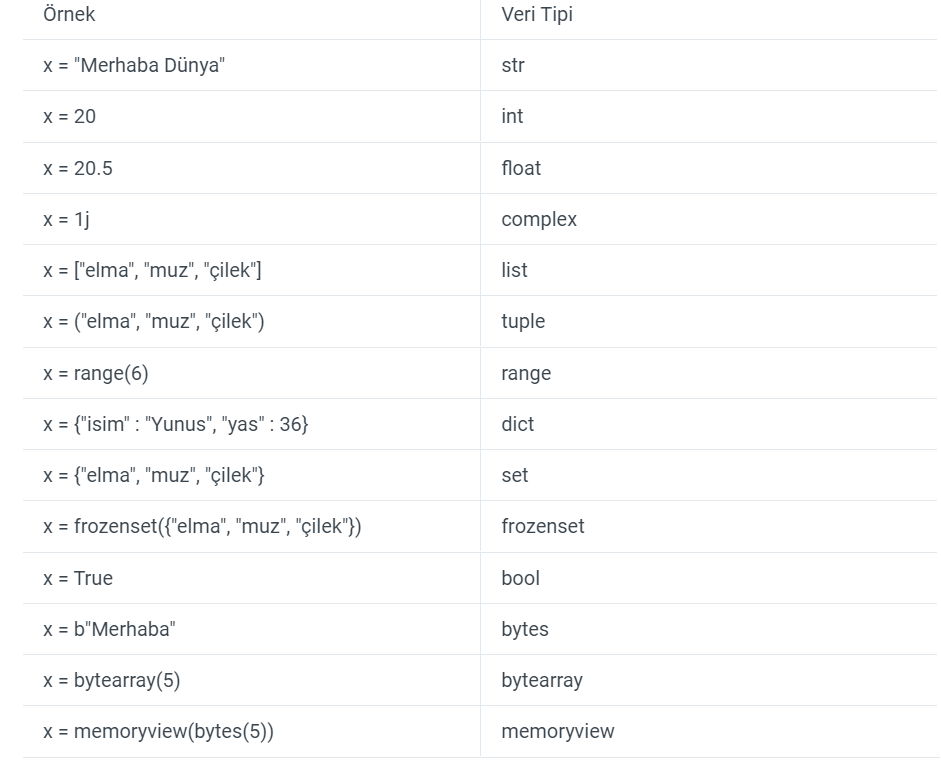
**Değişkenler (Veri Türleri)**

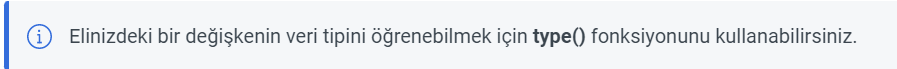
Hafızada (RAM) değişken değerlerini tutan alanlardır.

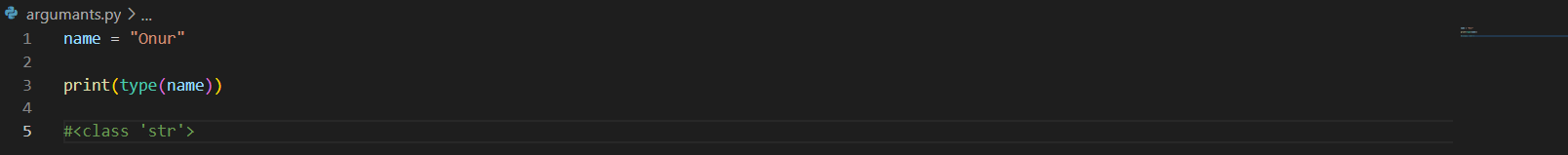
name= “ Onur ”



**Örnek Kullanımları**





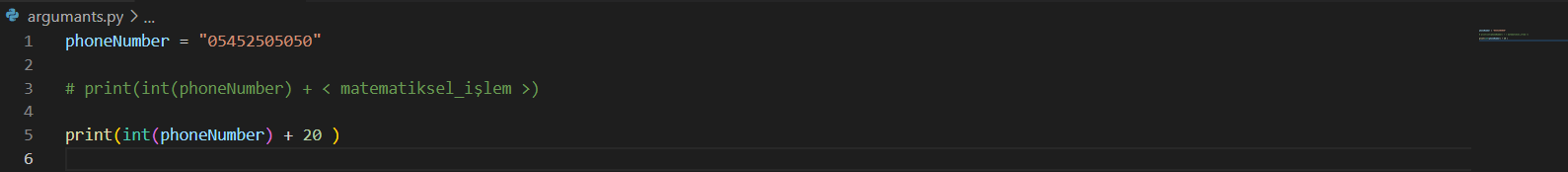


**Bilgi**: Phython dilinde değişken üretirken, değişken türü kullanmıyoruz.

Phython bunu kendisi otomatik olarak atıyor

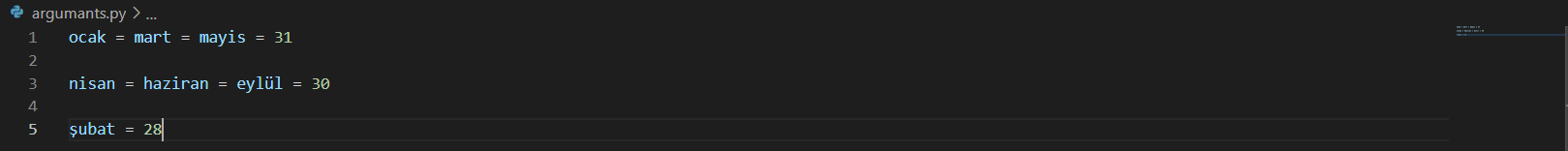


**BİLGİ**:



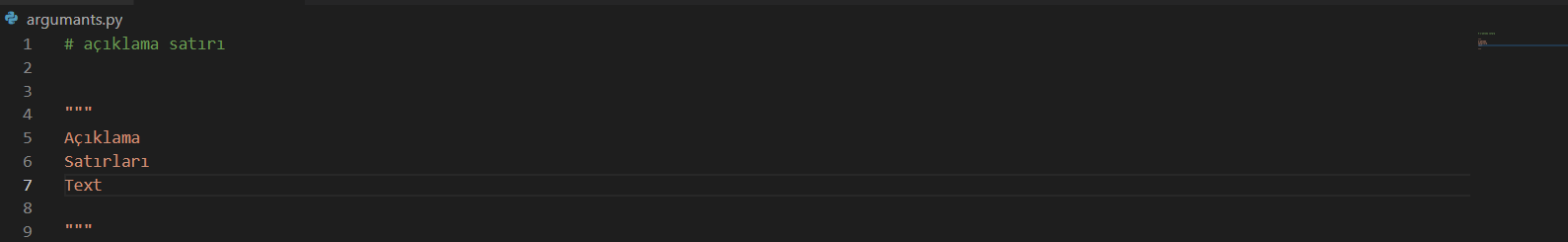
**Bilgi:** Python diline özgün bir değişken tanımlama şekli

Birden fazla değişkeni tek değere eşitlemek

i

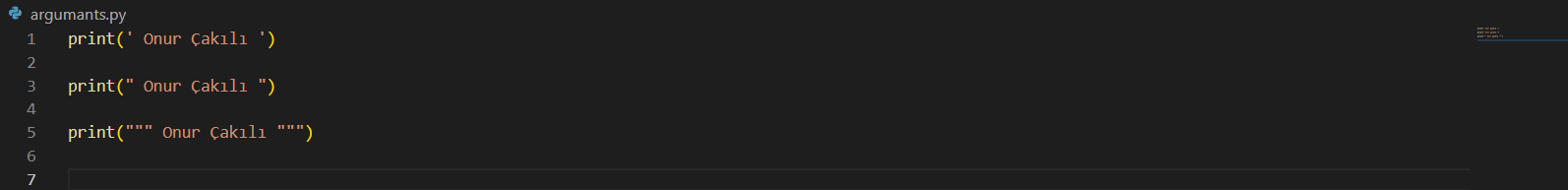
**Bilgi**: Python dilinde açıklama satırı (#) ile yapılır

Bir diğer yöntem şekliyle ise “”” text “””



**Bilgi:** Metinleri yazdırırken, çift tırnak veya tek veya üç tırnak içerisinde kullanmak python dilinde aynı şeyi ifade eder.

Bunların kullanımlarının kendilerine has özellikleri var. Bu yüzden 3 adet tırnak ifadesi kullanılıyor.



**Bilgi:** Değişkenleri adlandırırken sade isimler tercih etmeli ve yazarken okunaklı bir şekilde yazmalıyız.

**Örn**. faizOrani

İnvailed Syntax : Geçersiz Yazım

**String Değişkenlerin Metotları**

**Kaynak: (Python Tüm Metot ve Fonksiyonlar)**

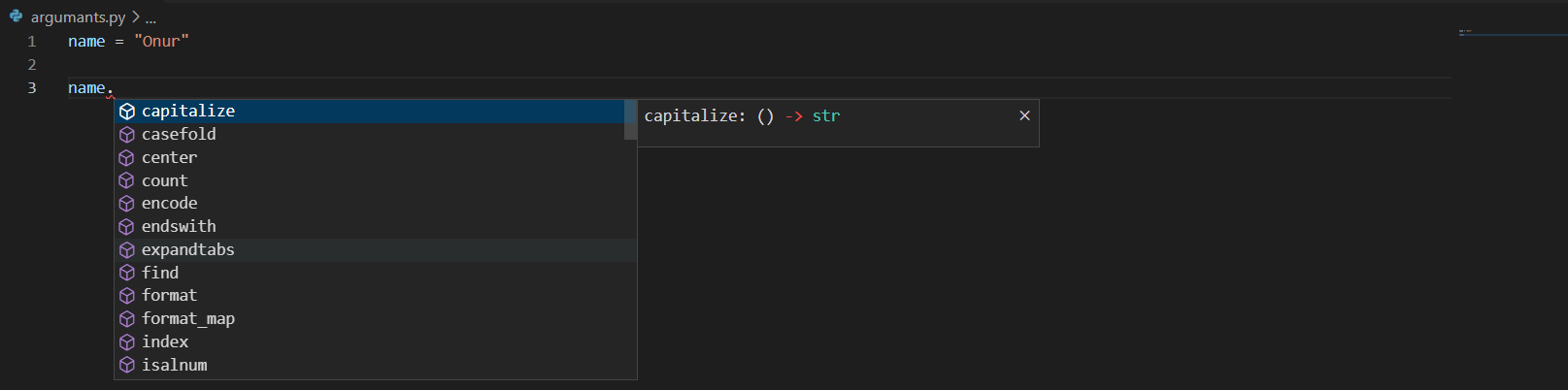
**https://python-global.com/python-metotlar/**

**Ön Bilgi: Metodlar kullanılırken sonlarına mutlaka parantez () kullanılmalıdır.**

Degisken.Metot()

**Ön Bilgi:** Metodun adını hatırlayamadığın veya hangisinin işine yarayacağını bulmak istediğin zamanlarda

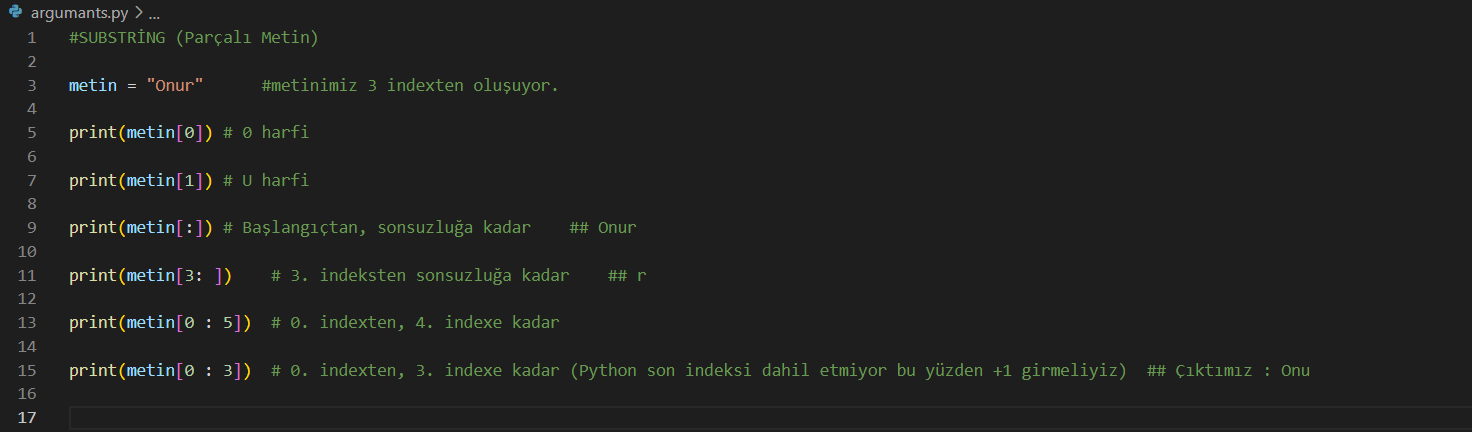
değişkenAdi . + space yaparak metodlara göz atabilirsin



1. **Substring (Parçalı Metin)**

Stringler, esasında char değişken türünün birleşiminden oluşan birer dizidir.

Örnek

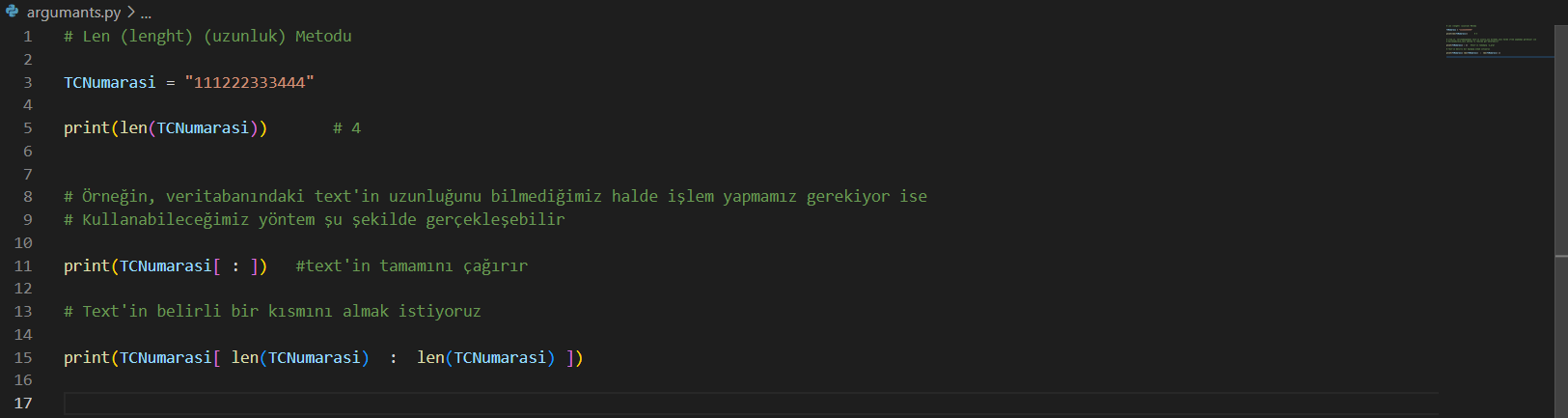


1. **Lenght (uzunluk)**

String ifadelerin kaç karakterden oluştuğunu geri döndüren metotdur.

Lenght metodu karakter bazlı sonuçlar verir. (index mantığı ile çalışmamaktadır)

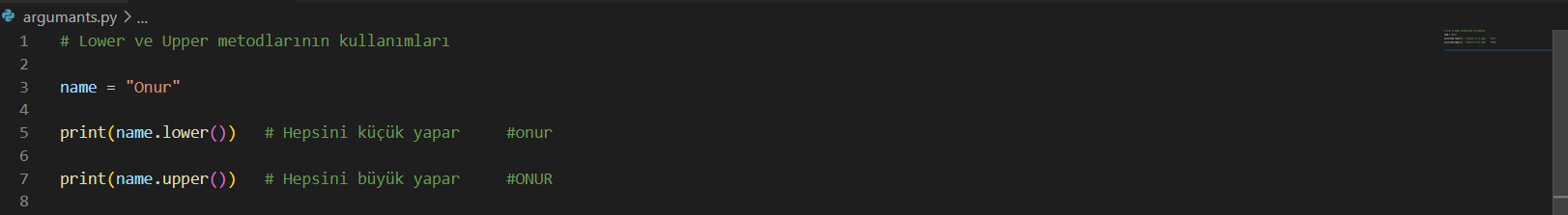
Space tuşları da karakter olarak işlem görmektedir



1. **Lower ve Upper**

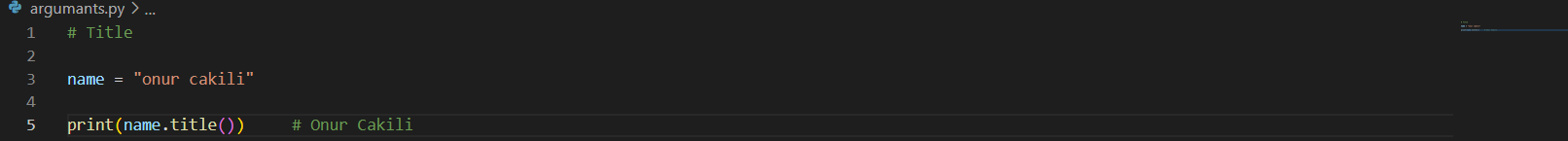
Lower, harfleri küçültmek

Upper ise harfleri büyültmek için kullanılan metodlardır.



1. **Title**

Title, metodu ise değişkenin sadece baş harfini büyük yapar



1. **Split ve Sprit Metodları**

split(), rsplit(), splitlines()

rsplit() sağdan sola doğru ayırma yapmaktadır

splitlines() ise satır bazlı ayırma yapmaktadır.

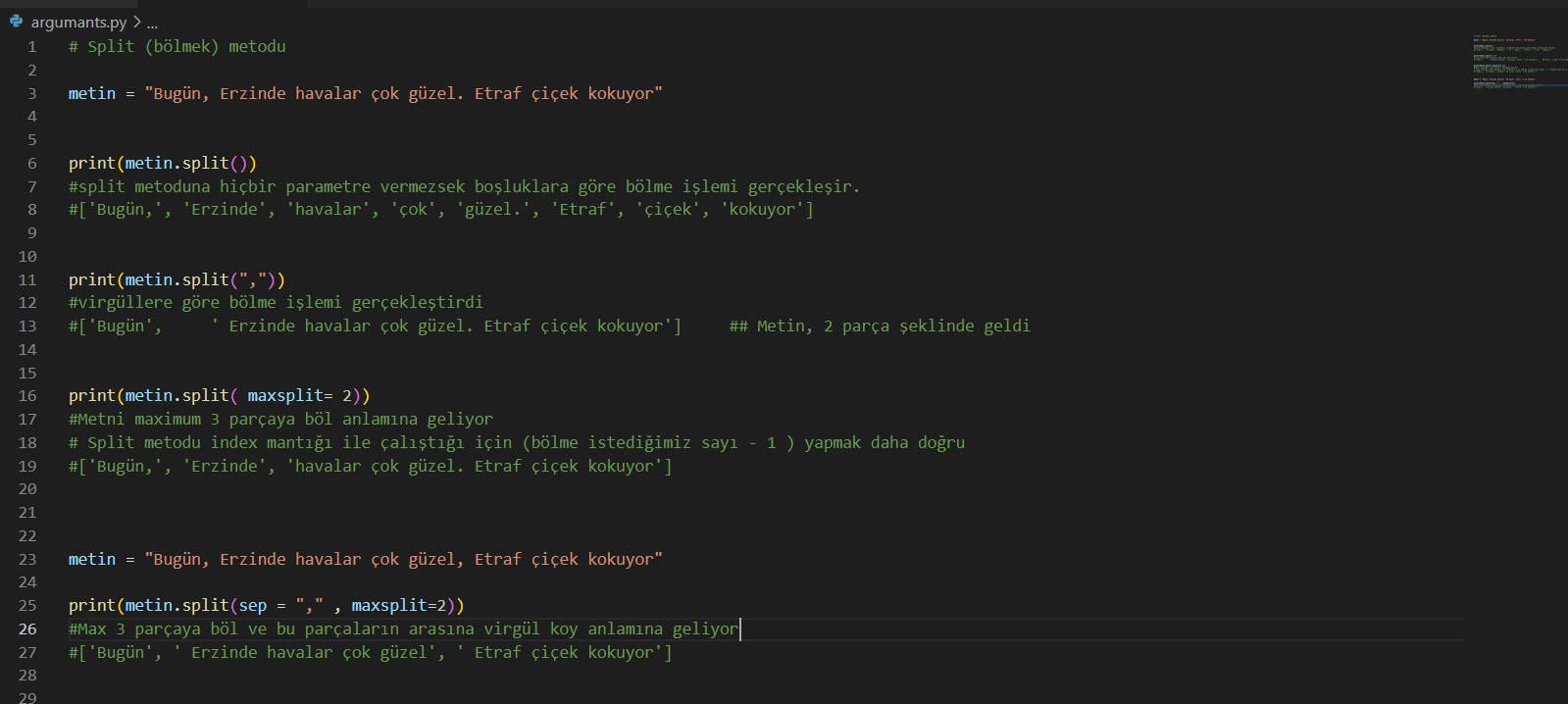
Split, İngilizcedeki bölmek, ayırmak kelimelerine karşılık gelmektedir.

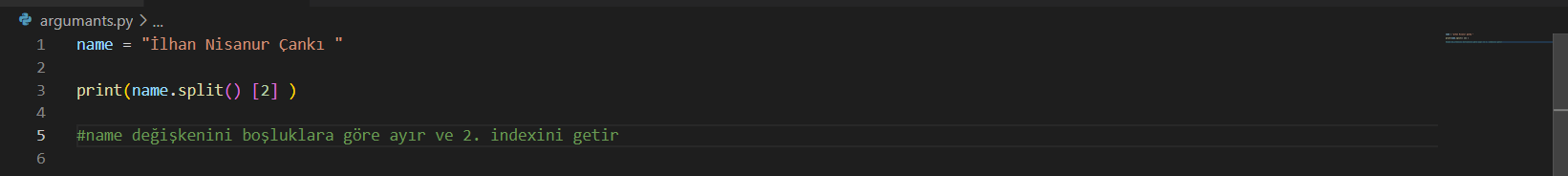
String ifadeleri bölmeye yaramaktadır.

Split metodu, index bazlı çalışmaktadır**.**

**Bilgi:** Sprit : Sadece fazladan boşlukları atmak için kullanılır ama bunu zaten splitte yapıyor o yüzden pek ihtiyaç yok gibi düşünebiliriz eğer illa buna gerek yoksa

#Eğer Split fonksiyona bir parametre girmezsek NULL yani boşluklara göre ayırma yapar



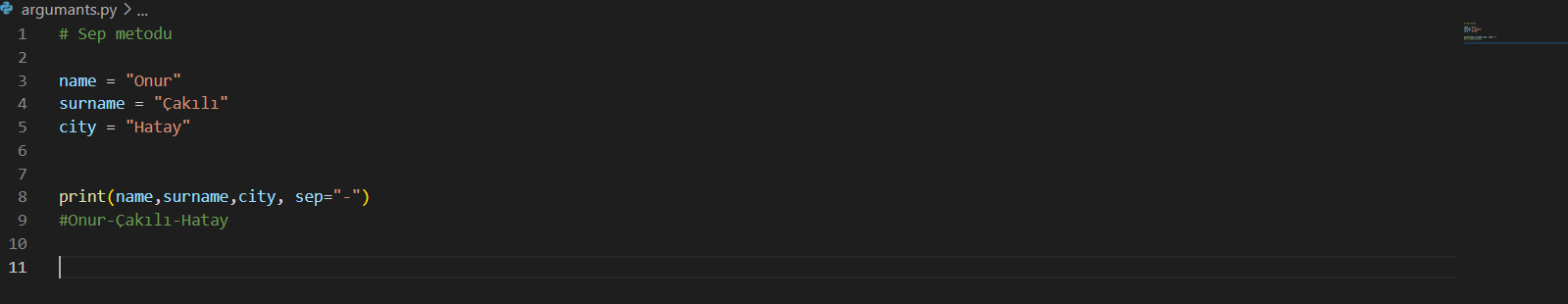




1. **Sep ve End metodları**

Sep metodu, İfadelerin arasına istenilen bir işareti vb. koymaya yarıyor.

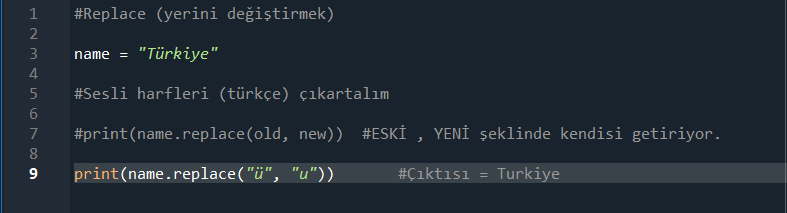
End metodu, İfadelerin sonuna istenilen bir işareti vb. koymaya yarıyor.





1. **Replace (Yer değiştirmek)**

String ifade içerisinde bulunan karakteri, belirlenen bir başka karakter ile değiştirmeye olanak sağlar

1. **Yıldız**

Yıldız metodu, belirlediğiniz değişken veya text’in karakter bazlı işlem görmesini sağlamaktadır.

Daha geniş kullanım şekilleri ise \*args veya \*\*kargs olarak bilinmektedir.

Veya (\*map(lambda x: x +5, liste)) gibi şekilleri de mevcuttur.

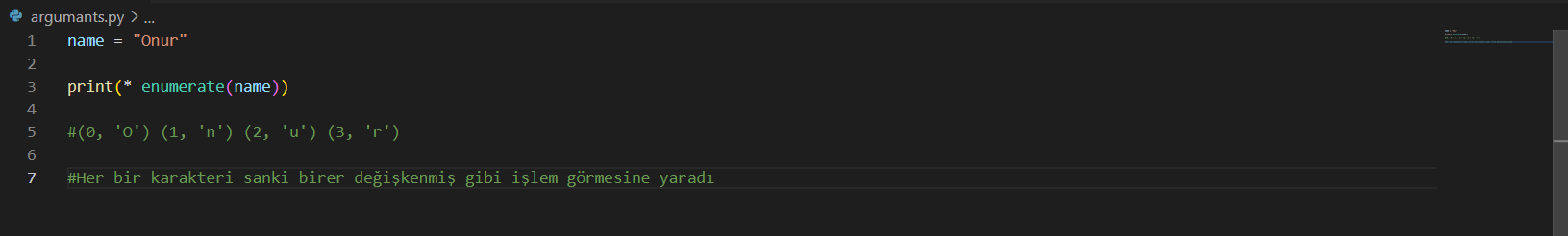
String ifadeler için ise şimdilik bu kadarı yeterlidir.



1. **Enumerate (numaralandırmak)**

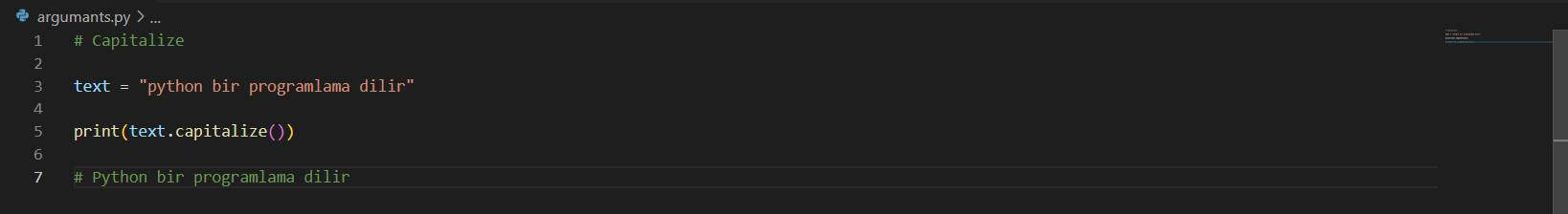
String içerisindeki karakterleri numaralandırmaya yarıyor





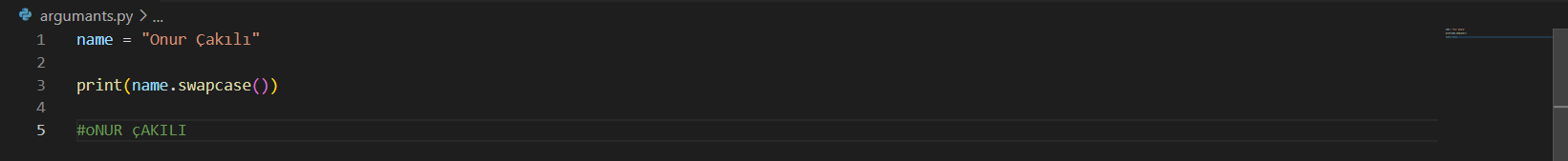
1. **Capitalize**

Capitalize, ilk kelimenin ilk harfini büyük yapmak için kullanılmaktadır.



1. **Swapcase (Takas Kutusu)**

Swapcase motodu, bir karakter dizisi içerisindeki büyük harfleri küçük, küçük harfleri ise büyük yapmaktadır.

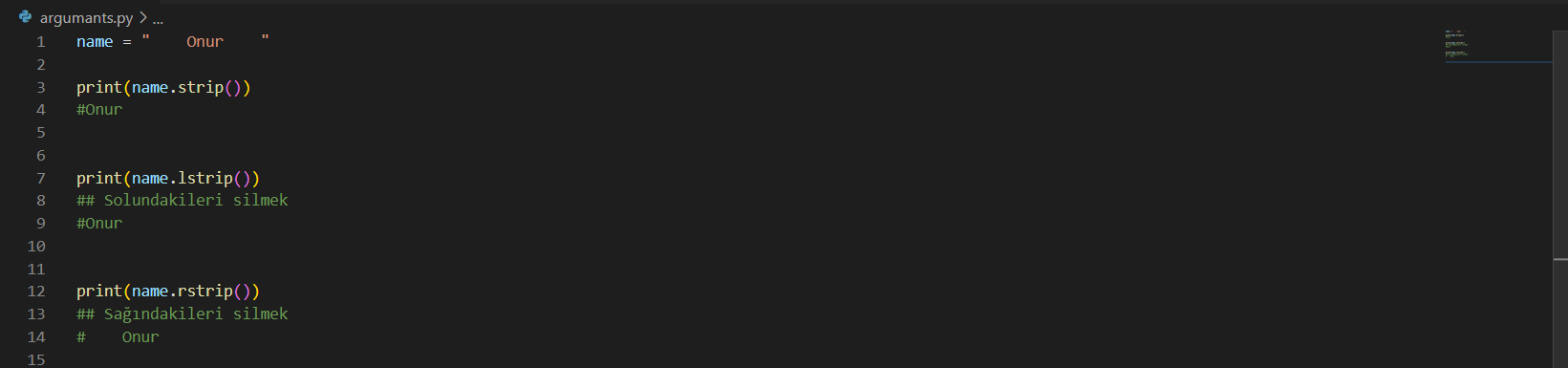


1. **Strip, lstrip, rstrip metodları**

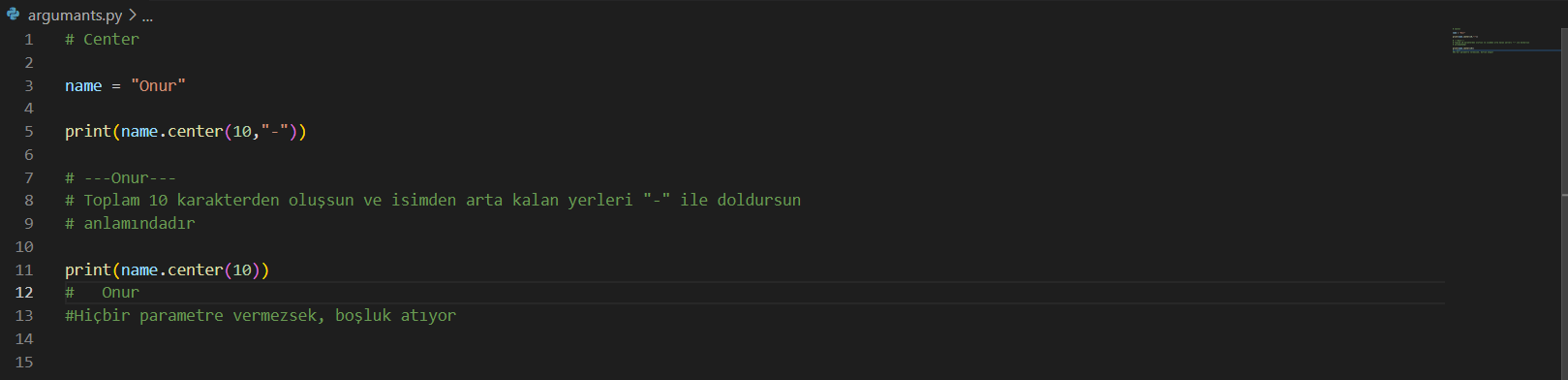
Strip() metodu, bir karakter dizisindeki istenmeyen şeyleri silmek için kullanılmaktadır.

Örneğin bir değişkenin sağ ve solunda boşlukları silmek için veya bir ifadeyi (-,<) silmek için de kullanabilirsiniz.

**NOT:** Split ve sprit metodları ile isimlerini karıştırmamaya dikkat et.

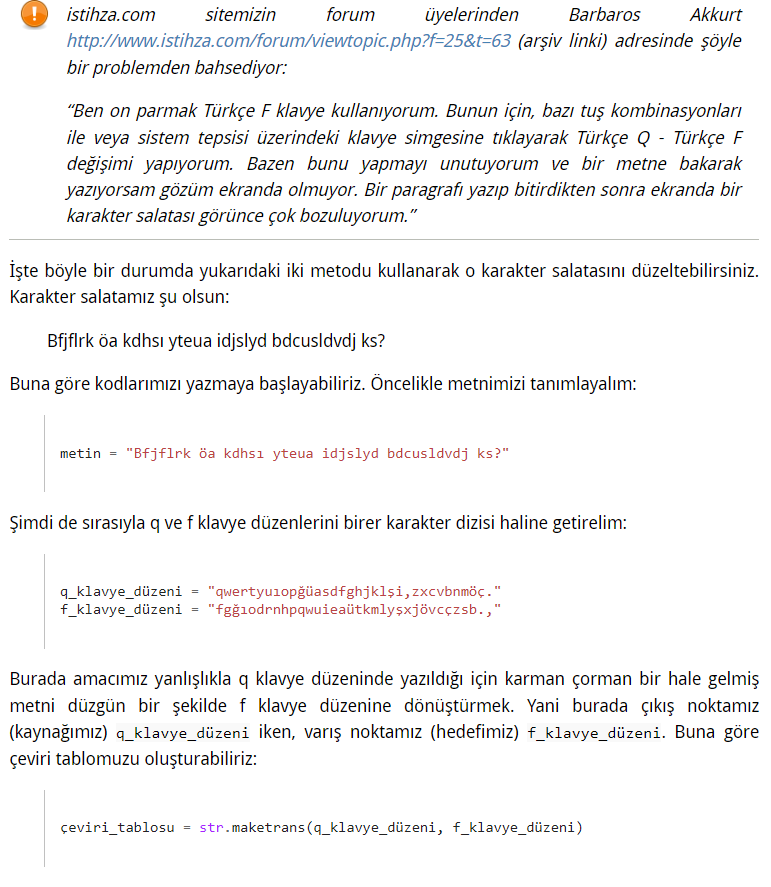


1. **Center (Merkez)**

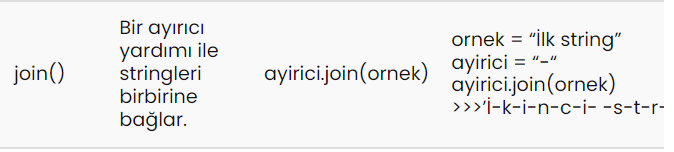


1. **Str.maketrans() ve translate()**





1. **Join**

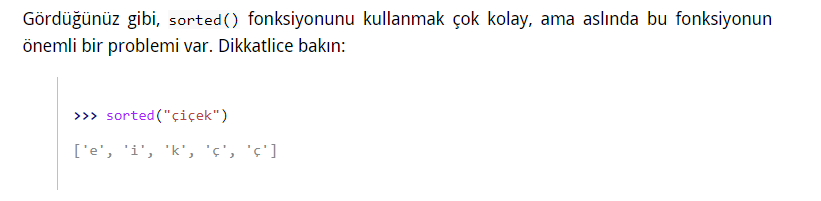


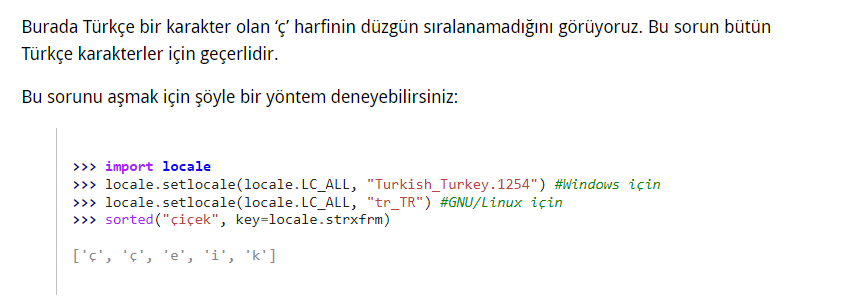
Genellikle

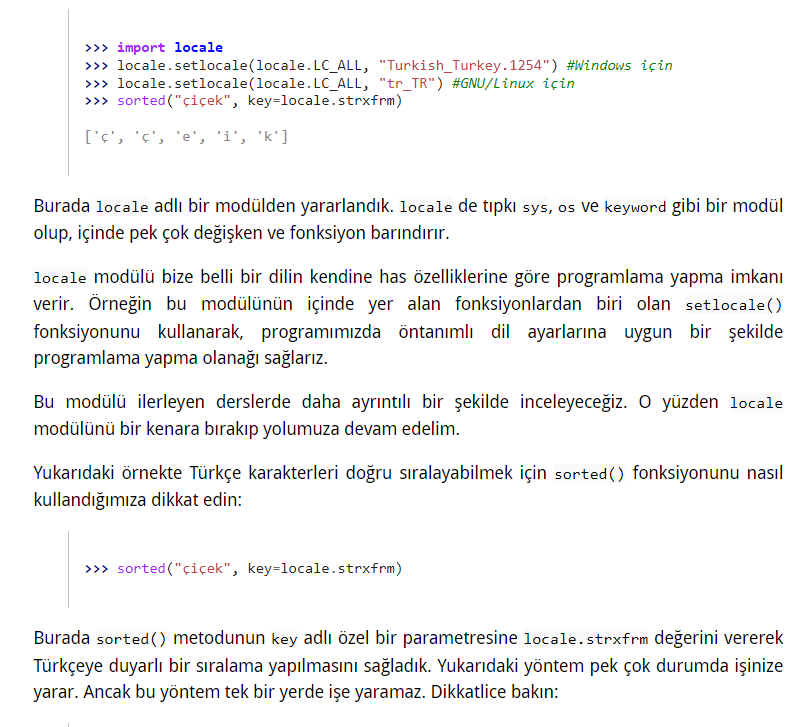
“ ”.join(degisken\_adi) şeklinde kullanılmaktadır.

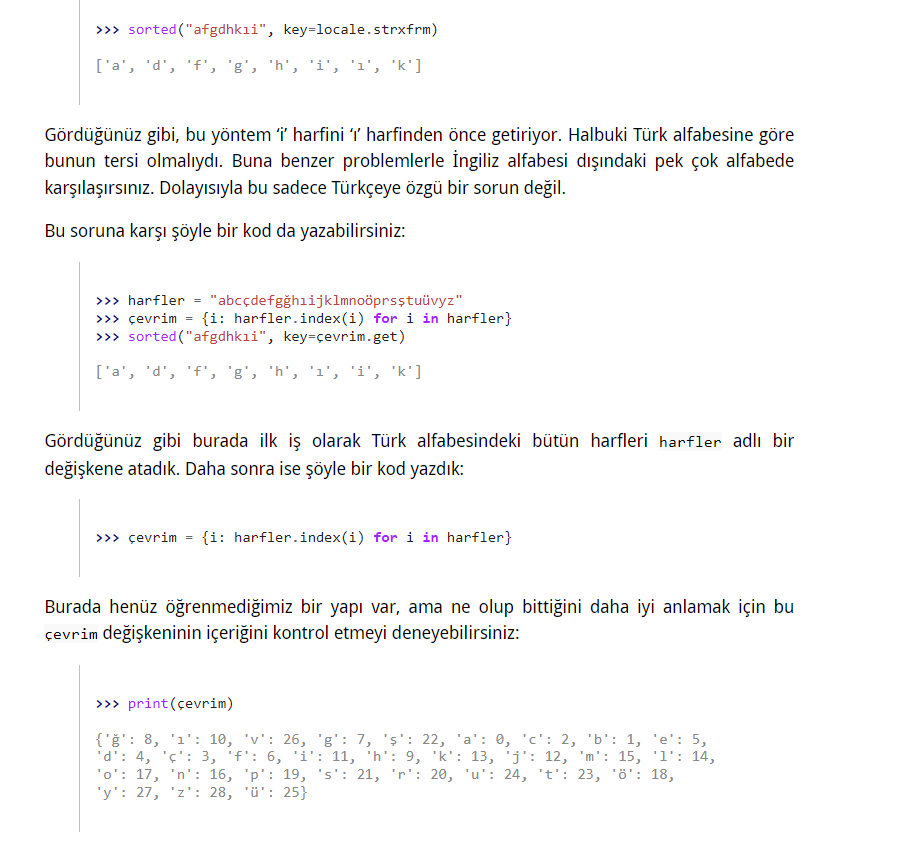
1. **Sorted**

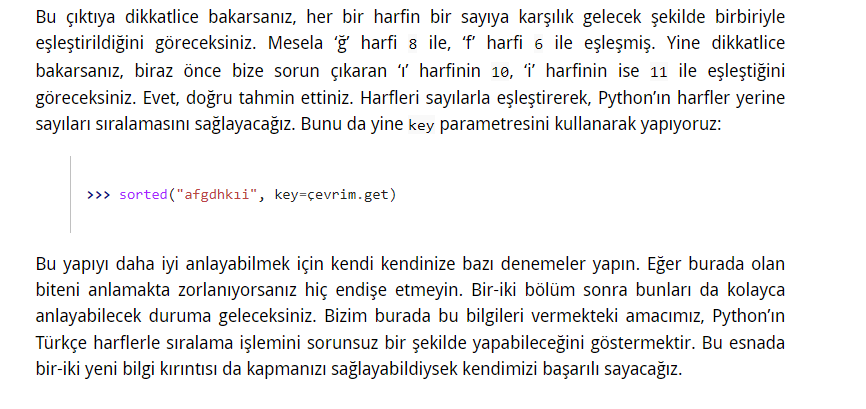
Python’da karakter dizilerinin öğelerine tek tek ulaşma, öğeleri dilimleme ve ters çevirmenin yanısıra, bu öğeleri alfabe sırasına dizmek de mümkündür. Bunun için sorted() adlı bir fonksiyondan yararlanacağız:











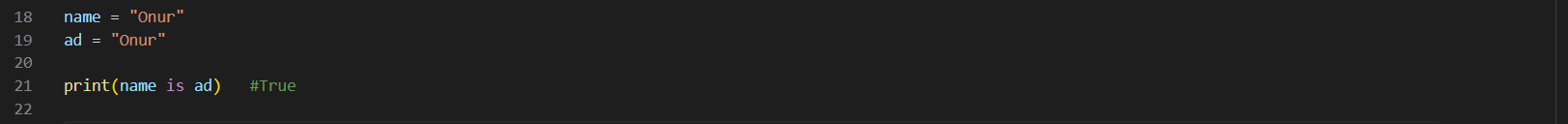
**String Test - Sorgulayıcı Metotlar ( identity / kimlik )**

Normal metodlar (upper,lower vb.) karakter dizileri üzerinde değişiklik yapan metotlardır.

Sorgulayıcı metodlar ise karakter dizileri üzerindeki durumu sorgulayan ve buna karşılık sadece True, False cevapları döndüren metodlardır.

1. is



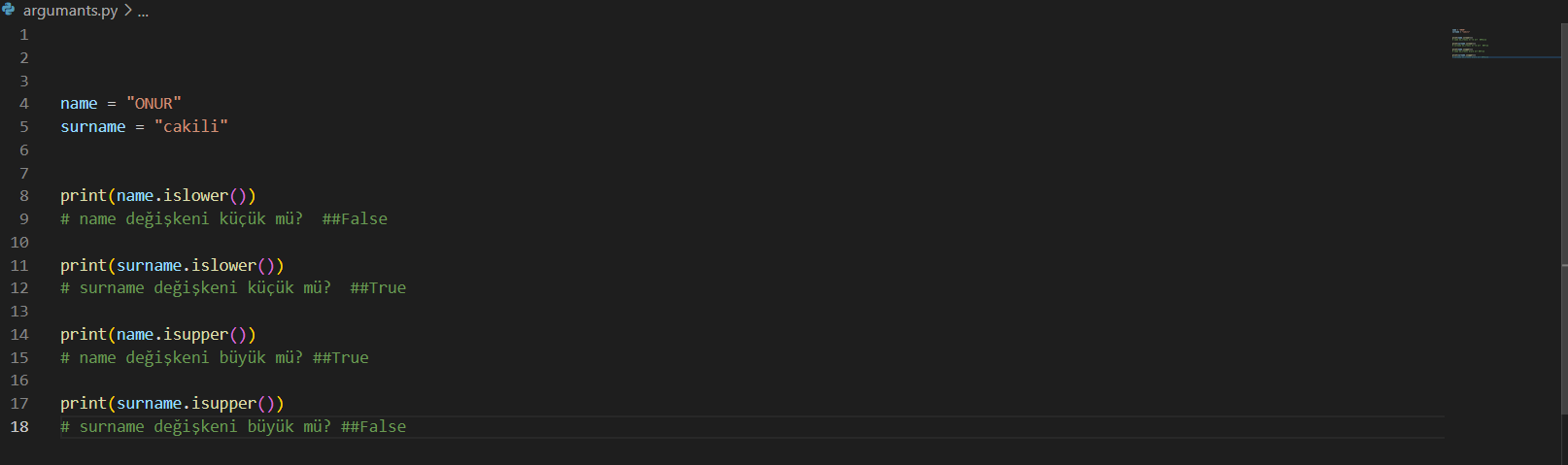


1. İslower() ve isupper()

Lower, karakter dizilerini küçük harfe – upper ise karakter dizilerini büyük harfe çevirmeye yarayan metodlardı.

İslower ve isupper ise bunların sorgularını yapan sorgulayıcı metodlardır.

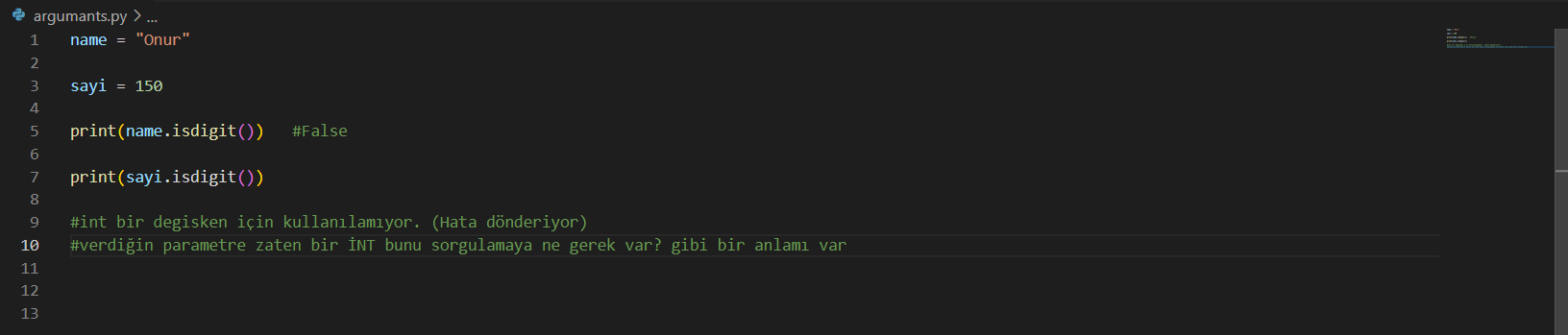
(Küçük mü? Veya büyük mü?) anlamlarında kullanılırlar.



1. İsdigit()

Digit : Rakam

Parametre olarak verilen şeyin bir rakam olup olmadığını sorgulamaktadır.



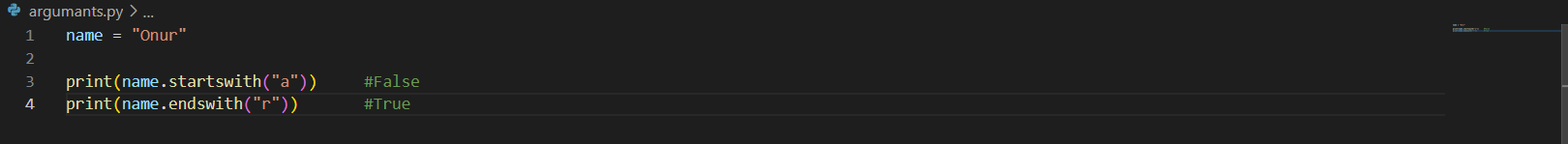
1. İsnumeric() - istitle() - isdecimal() - isspace() - isprintable()
2. Startswith() - endswith()

( - ile başlıyor)

( -ile bitiyor)

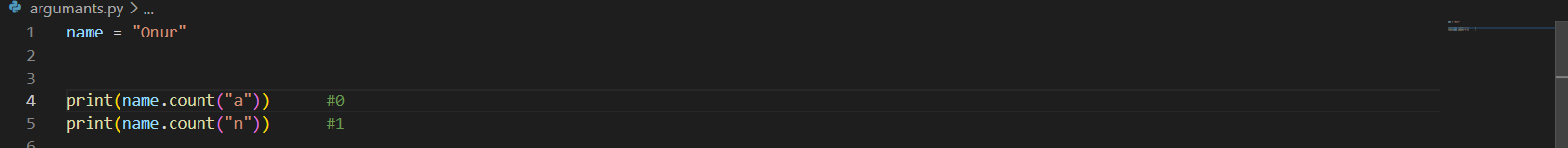
Startswith() metodu, karakter dizisinin verdiğimiz parametre ile başlayıp başlamadığını

Endswith() metodu ise karakter dizisinin verdiğimiz parametre ile bitip bitmediğini sorgulamaktadır



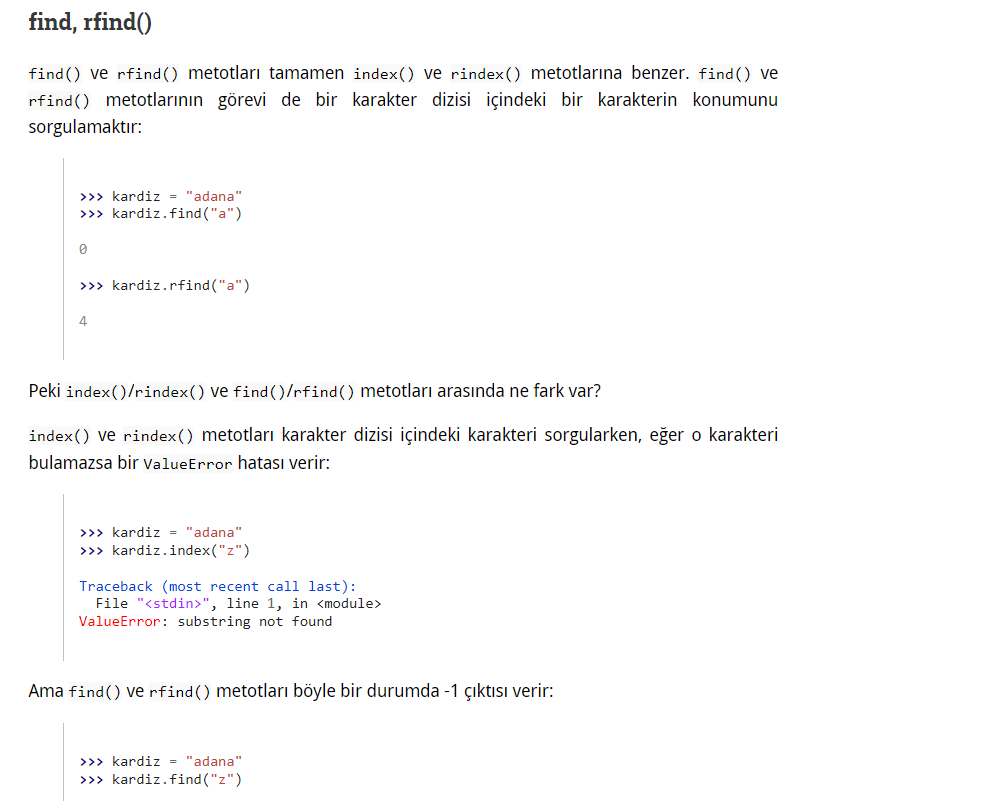
1. Count

Parametre olarak verilen ifadenin karakter dizesi içerisinde kaç adet olup olmadığını bize geri dönderen metod



1. Find() ve rfind() Metodları





**İnteger ( int ) Metodlar**

**BİLGİ :**

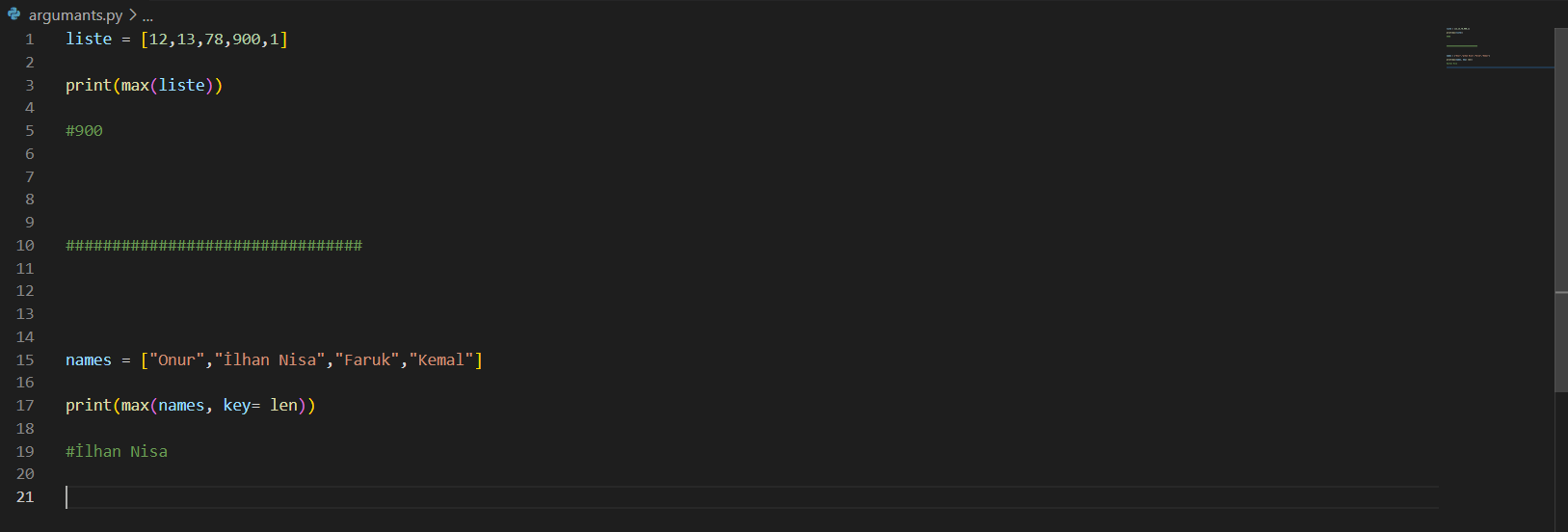
**Bilgisayarlar hakkında bilmemiz gereken en önemli bilgilerden biri şudur: Bilgisayarlar ancak ve ancak sayılarla işlem yapabilir.**

**Bilgisayarların işlem yapabildiği sayılar da onlu sistemdeki sayılar değil, ikili sistemdeki sayılardır. Yani 0’lar ve 1’ler.**

**Bilgisayar terminolojisinde bu 0’lar ve 1’lerden oluşan herbir basamağa ‘bit’ adı verilir. Yani ikili sayma sisteminde ‘0’ ve ‘1’ sayılarından herbiri 1 bit’tir. Mesela onlu sistemde 2 sayısının ikili sistemdeki karşılığı olan 10 sayısı iki bit’lik bir sayıdır. Onlu sistemdeki 100 sayısının ikili sistemdeki karşılığı olan 1100100 sayısı ise yedi bitlik bir sayıdır.**

1. **Max (maximum) ve Min (minimum)**

Parametre olarak verdiğimiz alanda en büyük sayıyı bulmaya yaran metod.

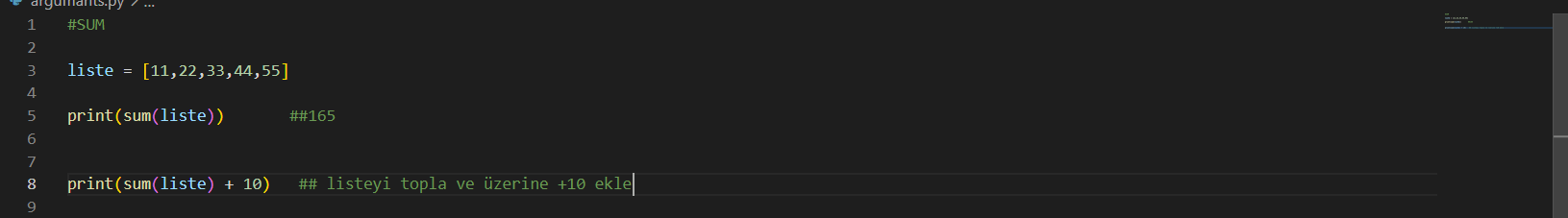


**Bilgi: key= ctrl + space**

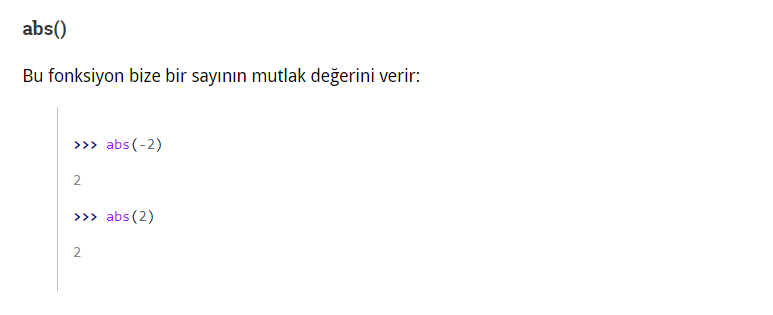
**Yaparak gereken metodları inceleyebilirsin**

1. **Sum (Toplam)**

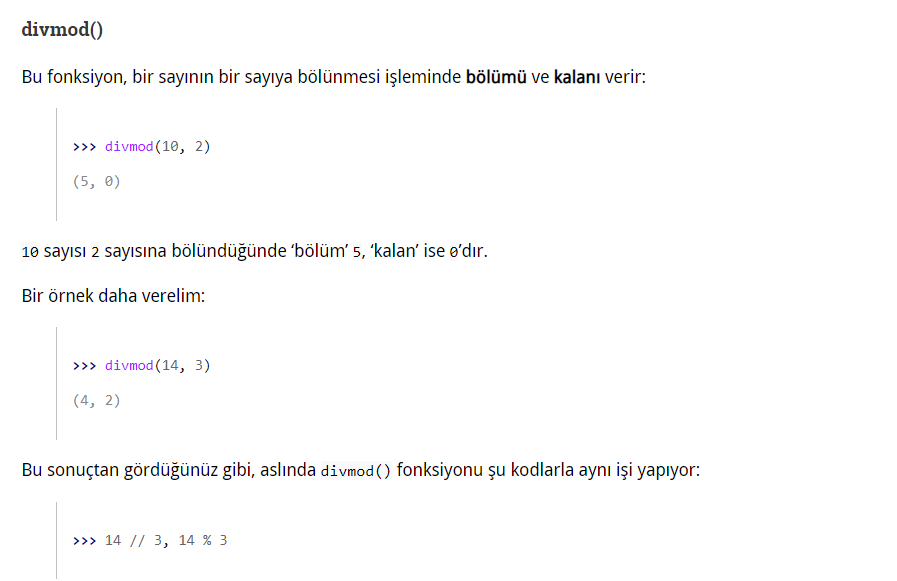
Parametre olarak verilenlerin (örn. Liste) tamamını toplamaya yarayan metod



1. **ABS (Mutlak Değer)**



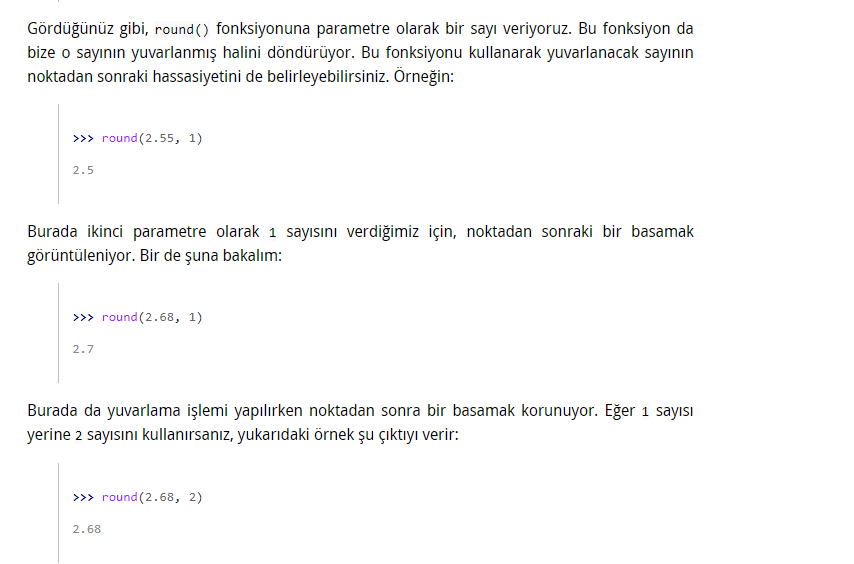
1. **Divmod (Bölümden Kalanı Verir) - Divied by and Mod**



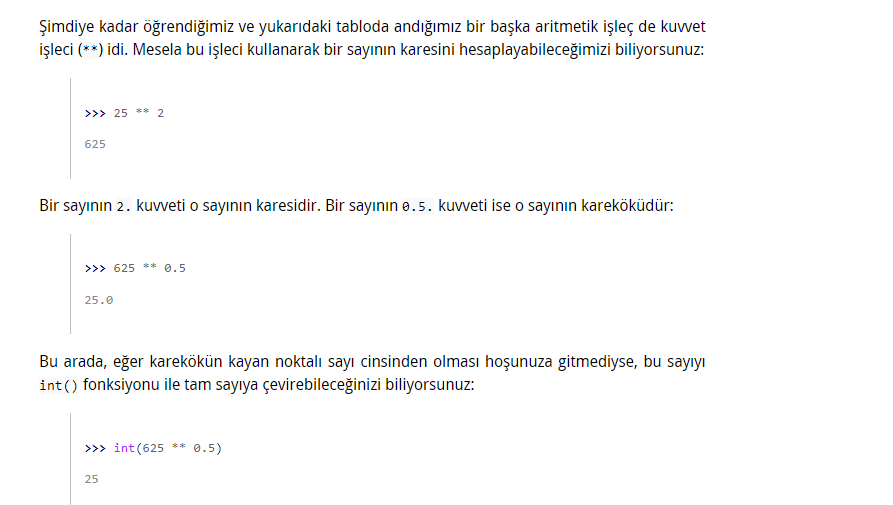
1. **Round**

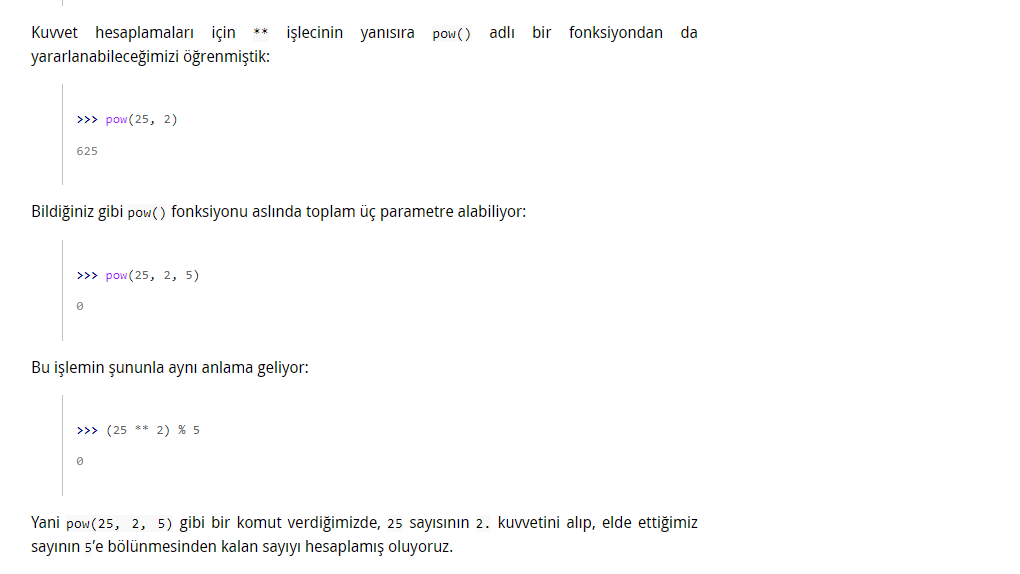
Round metodu, ondalıklı sayıları yuvarlamak için kullanılmaktadır.





1. **Pow metodu (üstlü sayılar)**





**Bilgi:**

**%40 = 0.40**

**X \* 0.5 = Kök(x)**

**Operatörler**

* Aritmetik (arithmetic) operatörler
* Atama (assignment) operatörleri
* Karşılaştırma (comparison) operatörleri
* Mantıksal (logical) operatörler
* Kimlik (identity) operatörleri
* Üyelik (membership) operatörleri
* Bit (bitwise) operatörleri

1. **Aritmetik Operatörler**



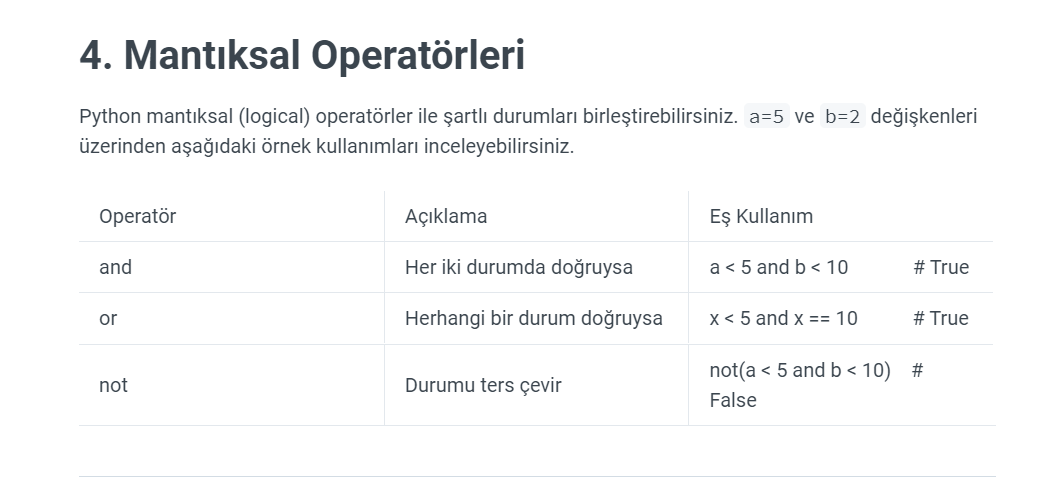
1. **Atama Operatörleri**



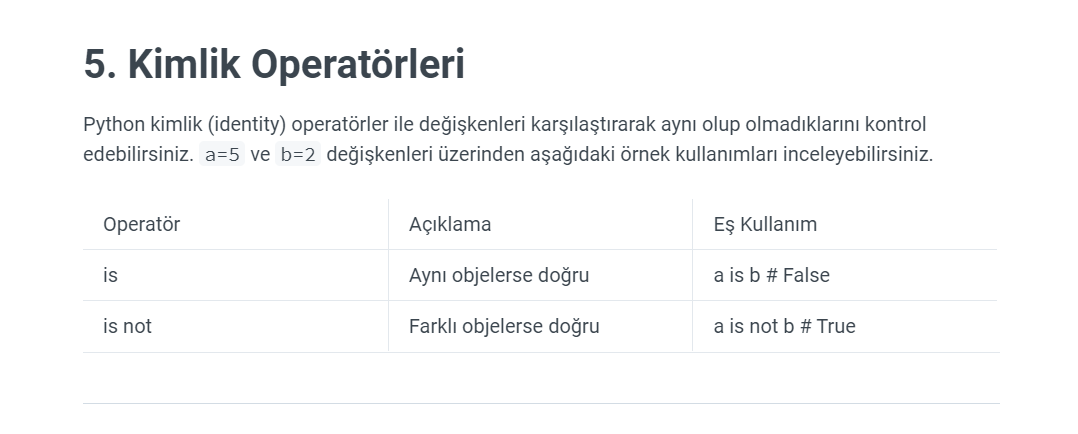
1. **Karşılaştırma Operatörleri**



1. **Mantıksal Operatörler**



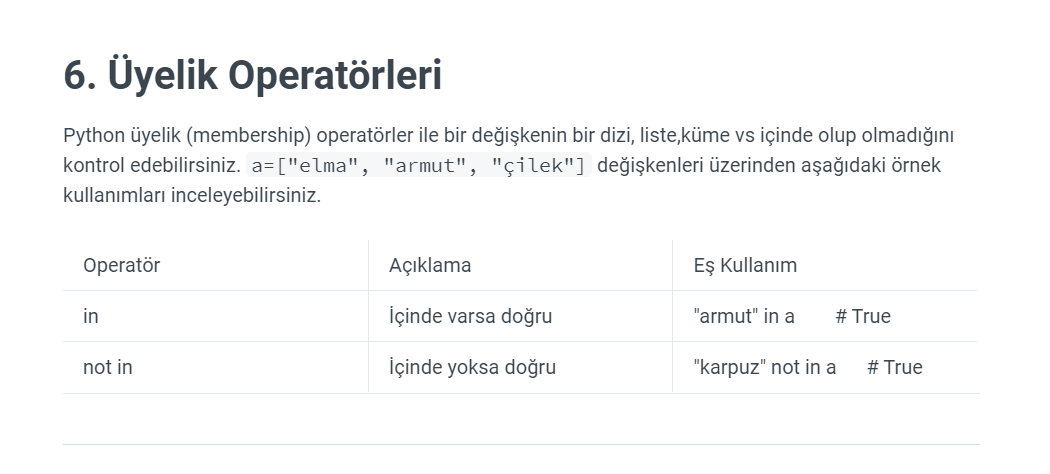
1. **Kimlik Operatörleri**



**İs : a ve b aynı**

**İs not = a ve b aynı değiş**

1. **Üyelik Operatörleri**



**İn : “armut” kelimesinin içerisinde “a” vardır**

**Not in : “armut” kelimesinin içerisinde “a” yoktur**