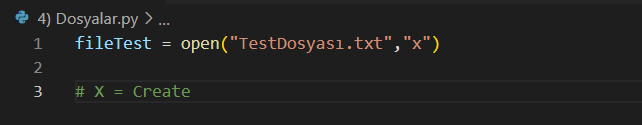
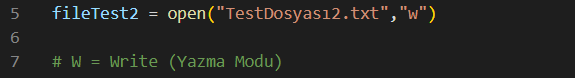
Dosya oluşturma Kipi



Dosya Yazdırma Kipi



Bilgi :Yukarıdaki komutu çalıştırdığınızda, o anda hangi dizin altında bulunuyorsanız o dizin içinde TestDosyasi2.txt adlı boş bir dosyanın oluşur

(Eğer belirlediğiniz isimde bir dosya zaten varsa o zaman da mevcut dosyayı silip tekrardan aynı isim ile bir dosya oluşturacaktır.)

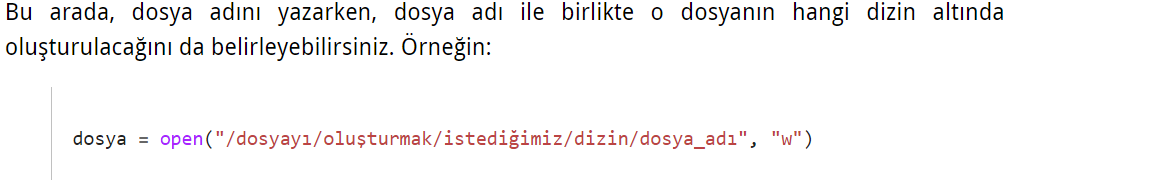
!!!

İçerisinde yazı olan bir dosyayı, tekrardan yazdırma “ w ” modunda açmaya çalışırsan

Yani > fileTest = open(“TestDosyası.txt”,”w”) yaparsan

İçerisinde yazılı olan tüm veriyi siler ve sıfırdan bir dosya oluşturur. Bu da veriyi kaybetmene sebep olur

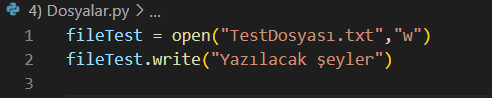
Bilgi:



Dosyaya Yazmak

Bir dosyayı, yukarıda gösterdiğimiz şekilde yazma kipinde açtığımız zaman, Python bizim için içi boş bir dosya oluşturacaktır.

Python programlama dilinde, open() fonksiyonu ile yazma kipinde açtığımız bir dosyaya bir veri yazabilmek için dosyaların write() adlı metodundan yararlanacağız.



Bilgi: Dosyaya Yazmak istiyorsak, dosyayı Yazdırma “ w ” kipinde açmamız gereklidir

Dosyayı Kapatmak

Üzerinde işlem yaptığımız dosyalar hafızada yer işgal ettiği için işimiz bittikten sonra dosyaları kapatmamız

Kaynaktaki hafızayı boşaltmamız gerekir. Bunun için Close metodundan yararlanıyoruz.



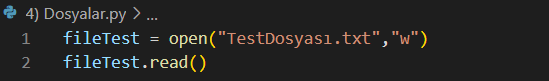
Dosya Okuma Kipi



“r” kipiyle açtığımız bir dosyayı okumak için ise read(), readline() ve readlines() adlı üç farklı metottan yararlanıyoruz

ÖNEMLİ EKSTRA BİLGİ !!!:

Dosyaya yazı yazmak istiyorsak W moduyla, dosyayı okumak istiyorsak R moduyla açmamız gerekiyor



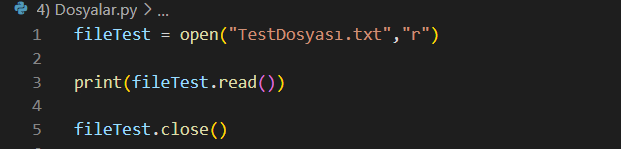
Yukarıdaki şekildeki gibi yazdırma modunda açarak okuma yapamıyoruz

En önemlisi, W modundaki open komutunu iki kez çalıştırınca zaten dosyanın içeriğini siliyordu.

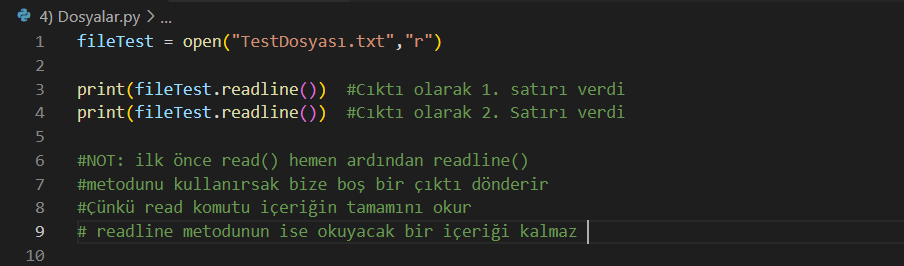
Bunu unutmuyoruz

Olması gereken …

Read() metodu , dosya içeriğinin tamamını okuyor

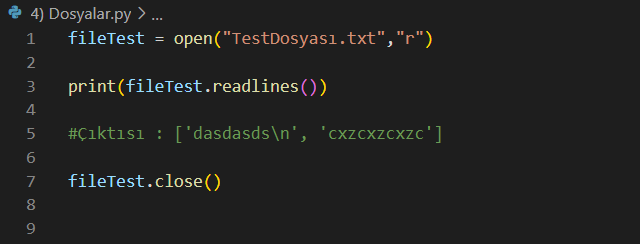


Readline() Metodu



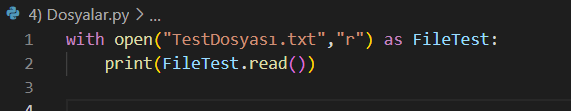
Readlines() Metodu

Readlines metodu, içeriği liste (arry) formatında bize verir

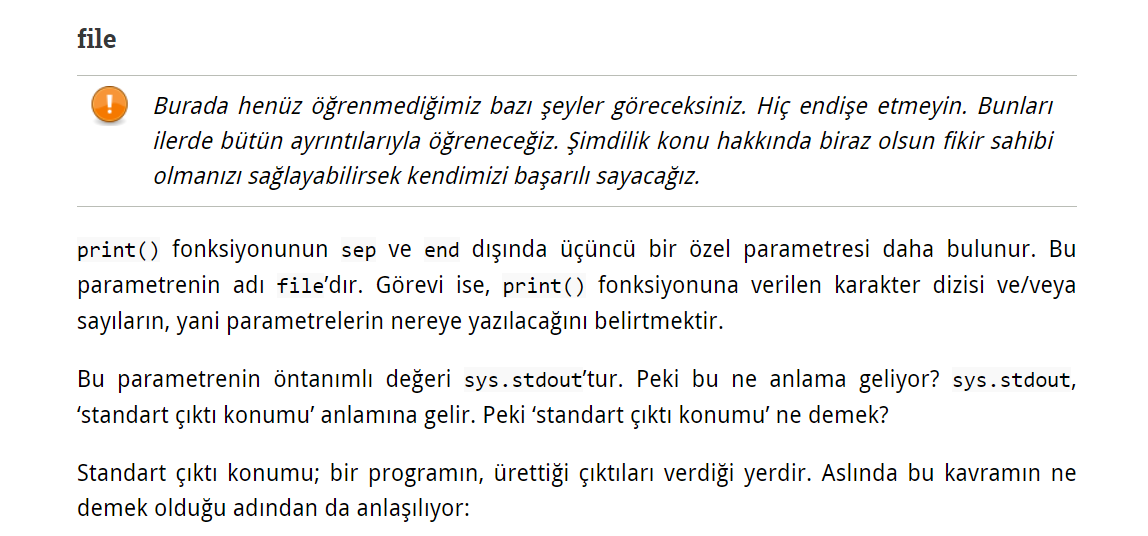


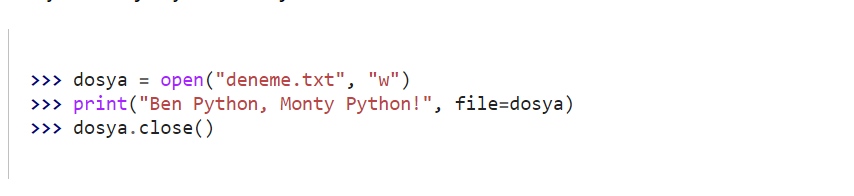
Otomatik Dosya Kapatmak

With deyimi ile birlikte dosyayı çalıştırırsak, Python satır sonunda otomatik olarak dosyayı kapatacaktır



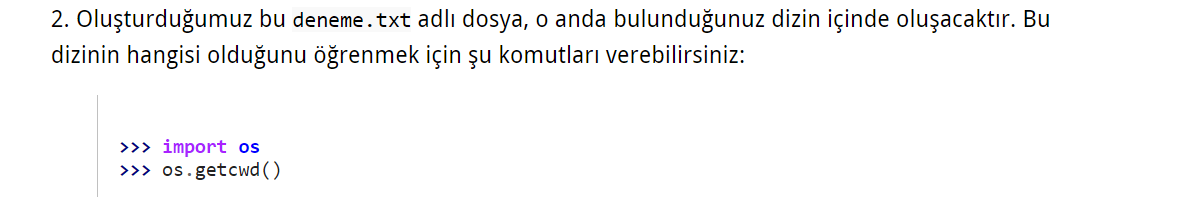
File Fonksiyonu





Artık standart çıktı komutumuz Console ekranı değil de file fonksiyonu ile belirtilen yer oldu

Bilgi: Dosya konumunu bulmak



DOSYA KONUMUNU KALICI OLARAK DEGİSTİRMEK (STANDART CIKTI KONUMUNU)

1.Adım

import sys

sys.stdout

Bu komut şuna benzer bir çıktı verir:

<\_io.TextIOWrapper name='<stdout>' mode='w' encoding='cp1254'>

2.Adım

Sys.sdout ‘ u istenilen dosya konumuna eşitliyoruz.

f = deneme.txt

sys.stdout = f

3.Adım

İmport os

Os.getcwd()

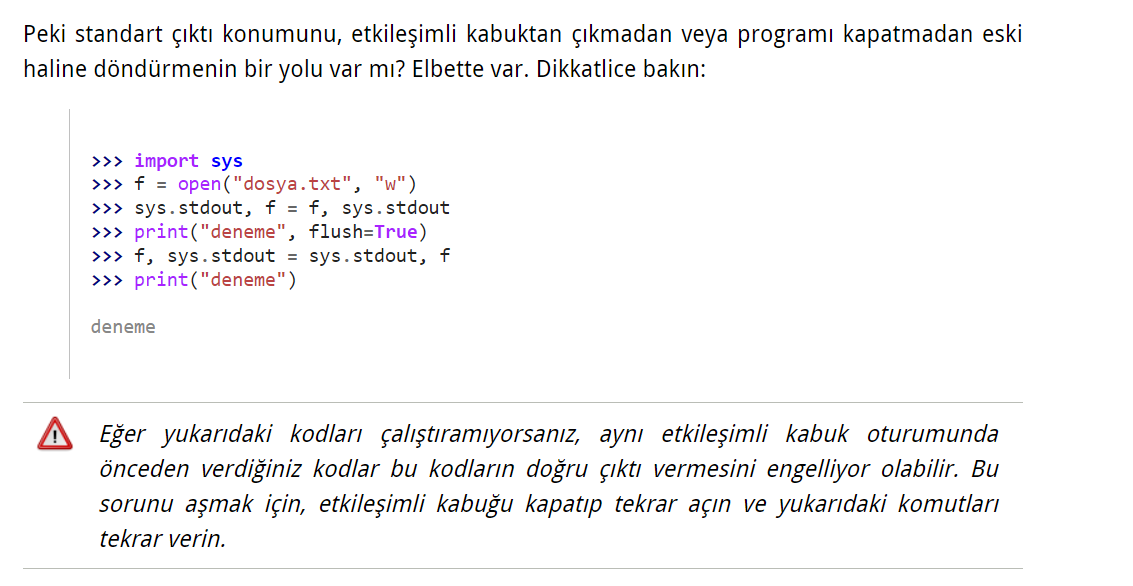
### <\_io.TextIOWrapper name='dosya.txt' mode='w' encoding='cp1254'>

Gördüğünüz gibi, özgün stdout çıktısındaki name=’<stdout>’ değeri name=’dosya.txt’ olmuş. Dolayısıyla artık bütün çıktılar dosya.txt adlı dosyaya gidiyor…

BİLGİ

: 

Tekrar Eski haline getirmek için ise



A,B = B,A

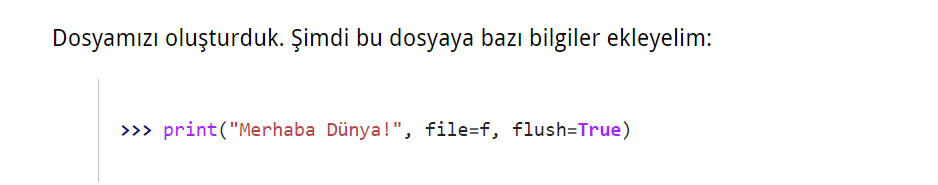
YER DEGİSTİRME DEGİSKİNLERDEKİ GİBİ MANTIGI

Flush fonksiyonu

Normalde programı Close edene kadar yazılan bilgiler dosyaya aktarılmıyordu sımdı flash kullanırsan A bile yazsan direkt dosyaya işliyor senin bitirip kapatmanı beklemıyor

anlık işlem görüyor

Kullanımı



(Normalde False olarak geliyor)