

Modüller ve Fonksiyonlar

Modül nedir?

Fonksiyon nedir?

Yerleşik Modüllere ait fonksiyonların kullanılması

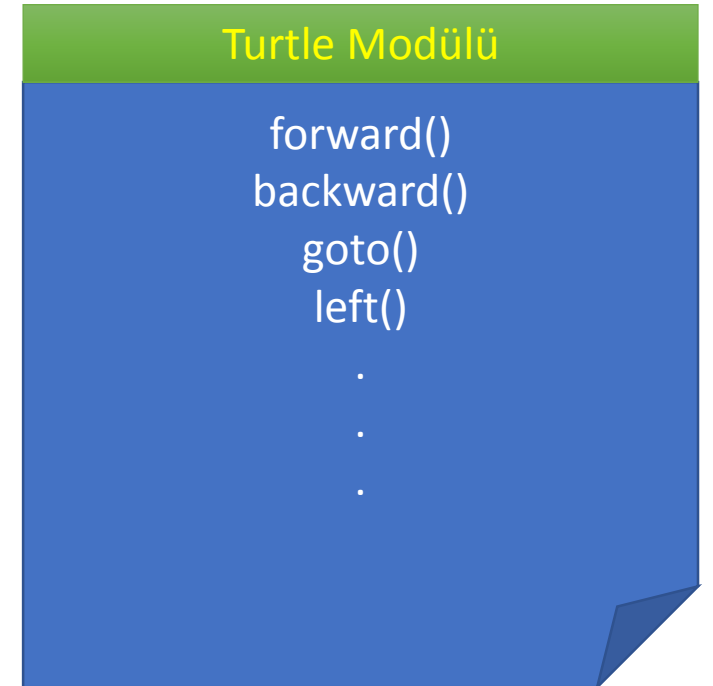
Modül Nedir?

Modüllerin, bazı işlevleri kolaylıkla yerine getirmemizi sağlayan birtakım fonksiyonları ve nitelikleri içinde barındıran araçlar olduğunu söyleyebiliriz. Modülleri python dosyasına import ederek içerisindeki fonksiyonları kullanabiliriz.

İki çeşit modül vardır.

1-Hazır Modüller

2-Kendi tanımladığımız modüller



PYTHON İLE BERABER GELEN HAZIR MODÜLLER

- Python ile beraber gelen bir çok modül bulunmaktadır. Biz bunlardan sadece aşağıdaki 3 modülü
- Kullanacağız.
- 1-Turtle Modülü
- 2-Time Modülü
- 3-Random Modülü

TURTLE MODÜLÜ

Turtle Modülü Nedir?

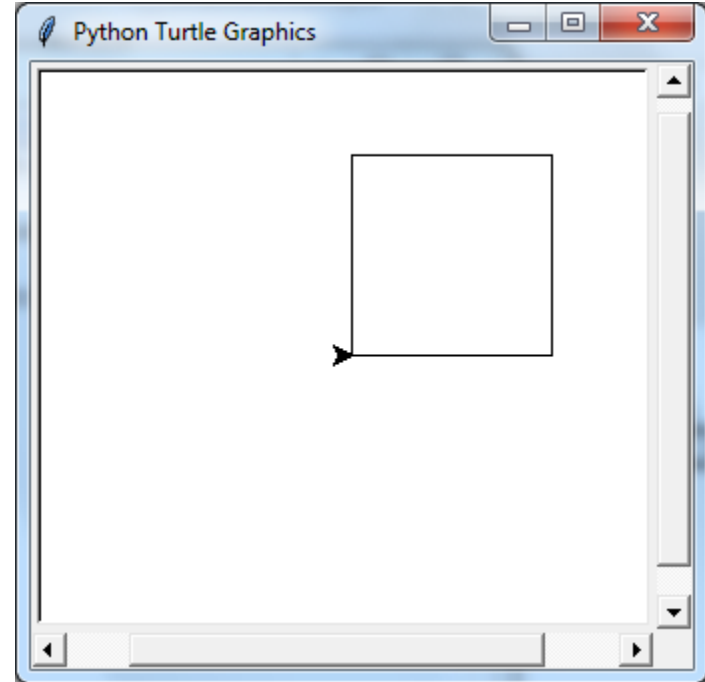
- Turtle modülünü Python sayfalarımıza import edebilir ve içerisinde yer alan çizim fonksiyonları ile çizimler yapabiliriz. Bu modülü import etmek için en önce aşağıdaki kod satırı yazılmalıdır.

```
from turtle import *
```

Turtle modülünü kullanarak 100 birimlik kare çizelim

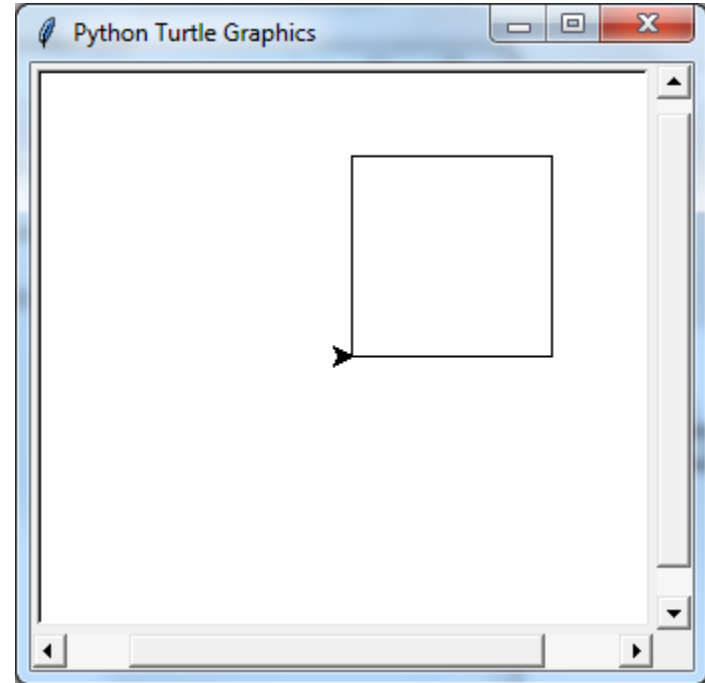
```
kareciz.py - C:/Python34/kareciz.py (3.4.4)
File Edit Format Run Options Window Help
from turtle import *

forward(100)      #100 birimlik ileri çizgi çiz
left(90)         #90 derece sola dön
forward(100)     #100 birimlik ileri çizgi çiz
left(90)         #90 derece sola dön
forward(100)     #100 birimlik ileri çizgi çiz
left(90)         #90 derece sola dön
forward(100)     #100 birimlik ileri çizgi çiz
left(90)         #90 derece sola dön
|
Ln: 11 Col: 0
```

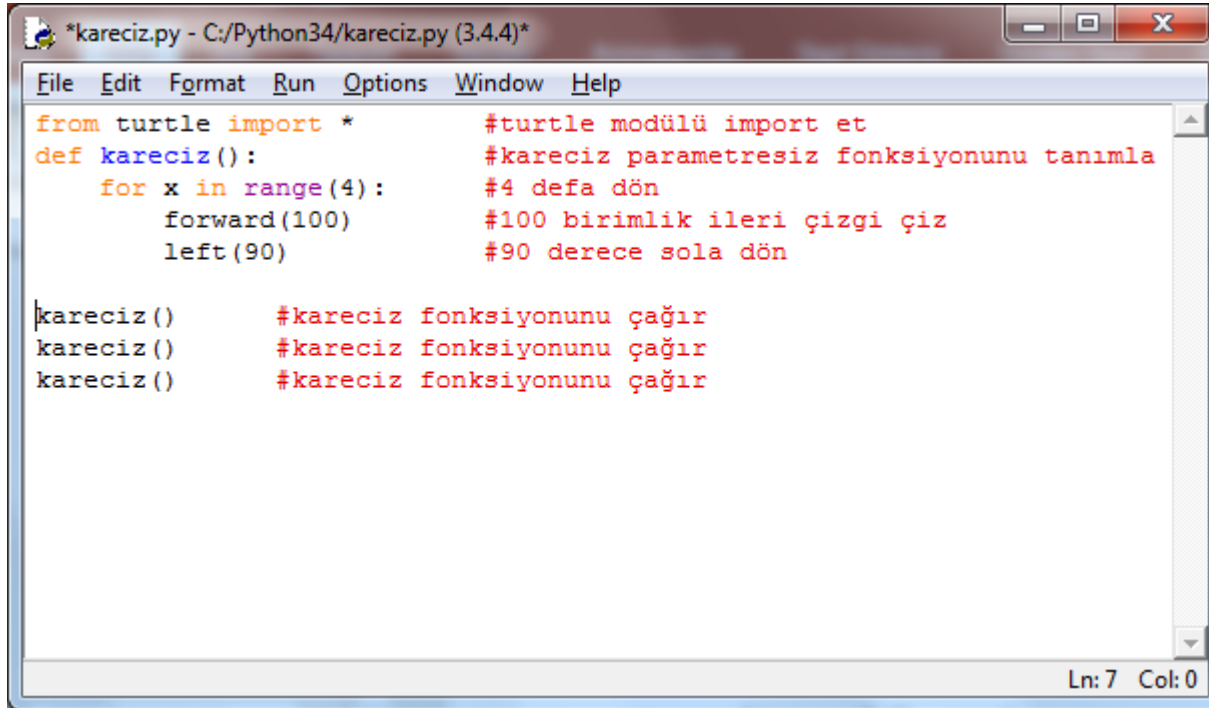


Turtle modülünü kullanarak 100 birimlik kare çizelim(for döngüsü ile)

```
kareciz.py - C:/Python34/kareciz.py (3.4.4)
File Edit Format Run Options Window Help
from turtle import *
for x in range(4): #4 defa dön
    forward(100)    #100 birimlik ileri çizgi çiz
    left(90)        #90 derece sola dön
|
Ln: 5 Col: 0
```



Turtle modülünü kullanarak 100 birimlik kare çizelim(fonksiyon ve for döngüsü kullanarak)



```
*kareciz.py - C:/Python34/kareciz.py (3.4.4)*
File Edit Format Run Options Window Help
from turtle import *          #turtle modülü import et
def kareciz():                #kareciz parametresiz fonksiyonunu tanımla
    for x in range(4):        #4 defa dön
        forward(100)          #100 birimlik ileri çizgi çiz
        left(90)              #90 derece sola dön

kareciz()                     #kareciz fonksiyonunu çağır
kareciz()                     #kareciz fonksiyonunu çağır
kareciz()                     #kareciz fonksiyonunu çağır

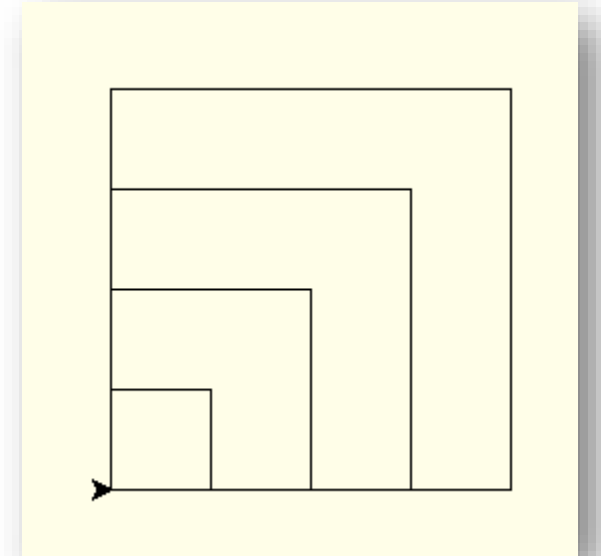
Ln: 7 Col: 0
```


Parametre olarak gönderilen değere göre kare çizen programı yazalım

```
*kareciz.py - C:/Python34/kareciz.py (3.4.4)*
File Edit Format Run Options Window Help
from turtle import *           #turtle modülü import et
def kareciz(birim):           #kareciz parametrelili fonksiyonunu tanımla
    for x in range(4):        #4 defa dön
        forward(birim)        #Gelen paramere kadar ileri çizgi çiz
        left(90)              #90 derece sola dön

kareciz(50)                   #kareciz fonksiyonunu çağır 50 parametresini gönder
kareciz(100)                  #kareciz fonksiyonunu çağır 100 parametresini gönder
kareciz(150)                  #kareciz fonksiyonunu çağır 150 parametresini gönder
kareciz(200)                  #kareciz fonksiyonunu çağır 200 parametresini gönder

Ln: 10 Col: 49
```



Girilen kenar sayısına göre 100 ve X birimlik ilgili şekli çizen programı yazalım

```
*sekilciz.py - C:/Python34/sekilciz.py (3.4.4)*
File Edit Format Run Options Window Help
from turtle import *      #turtle modülü import et
def sekilciz(kenarsayisi): #şekilciz parametrelili fonksiyonunu tanımla
    disAci=360/kenarsayisi #dönme açısını hesapla
    for x in range(kenarsayisi): #kenar sayısı kadar dön
        forward(100)          #100birim kadar ileri çizgi çiz
        right(disAci)         #hesaplanan derece kadar sağa dön

sekilciz(3)      #Üçgen çizer
clear()         #Pencereyi temizler
sekilciz(4)     #Kare çizer
clear()        #Pencereyi temizler
sekilciz(5)    #beşgen çizer
clear()       #Pencereyi temizler
sekilciz(6)   #6 gen çizer
clear()      #Pencereyi temizler
sekilciz(7)  #7gen çizer
clear()     #Pencereyi temizler

Ln: 18 Col: 0
```

```
*sekilciz.py - C:/Python34/sekilciz.py (3.4.4)*
File Edit Format Run Options Window Help
from turtle import *      #turtle modülü import et
def sekilciz(kenarsayisi,birim): #şekilciz parametrelili fonksiyonunu tanımla
    disAci=360/kenarsayisi #dönme açısını hesapla
    for x in range(kenarsayisi): #kenar sayısı kadar dön
        forward(birim)          #100birim kadar ileri çizgi çiz
        right(disAci)         #hesaplanan derece kadar sağa dön

sekilciz(3,50)  #50 birimlik Üçgen çiz
clear()        #Pencereyi temizler
sekilciz(4,100)#100 birimlik Kare çiz
clear()       #Pencereyi temizler
sekilciz(5,150)#150 birimlik beşgen çiz
clear()      #Pencereyi temizler
sekilciz(6,100)#100 birimlik 6 gen çiz
clear()     #Pencereyi temizler
sekilciz(7,100)#100 birimlik 7gen çiz
clear()    #Pencereyi temizler

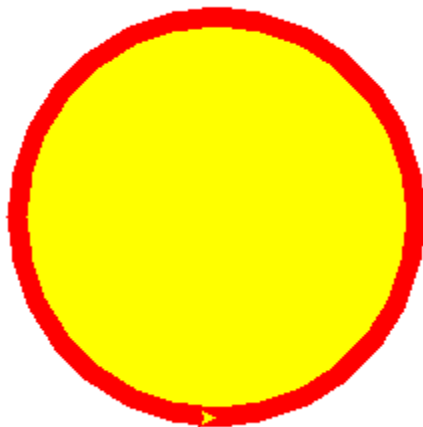
Ln: 16 Col: 30
```

Şekilleri renklendirelim

```
*dairerenklendir.py - C:/Python34/dairerenklendir.py (3.4.4)*
File Edit Format Run Options Window Help
from turtle import * #turtle modülünü import et

pensize(10) #kalem kalınlığını 10 birim yap
color("red","yellow") #color(cizgirenk,dolgurenk) renk tanımla
begin_fill() #renklendirme başla
circle(100) #100 birimlik daire çiz
end_fill() #renklendirme bitir

Ln: 3 Col: 24
```



```
sekilciz.py - C:/Python34/sekilciz.py (3.4.4)
File Edit Format Run Options Window Help
from turtle import * #turtle modülü import et

def sekilciz(kenarsayisi,birim): #şekilciz parametrelili fonksiyonunu tanımla
    disAci=360/kenarsayisi #dönme açısını hesapla
    pensize(5) #kalem boyunu 5 birim yap
    color("blue","gray") #çizgi rengini mavi, dolgu rengini gri yap
    begin_fill() #boyama başla
    for x in range(kenarsayisi): #kenar sayısı kadar dön
        forward(birim) #100birim kadar ileri çizgi çiz
        right(disAci) #hesaplanan derece kadar sağa dön
    end_fill() #boyama bitir

#*****
sekilciz(3,50) #50 birimlik Üçgen çiz
sekilciz(4,100) #100 birimlik Kare çiz
sekilciz(5,150) #150 birimlik beşgen çiz

Ln: 11 Col: 37
```

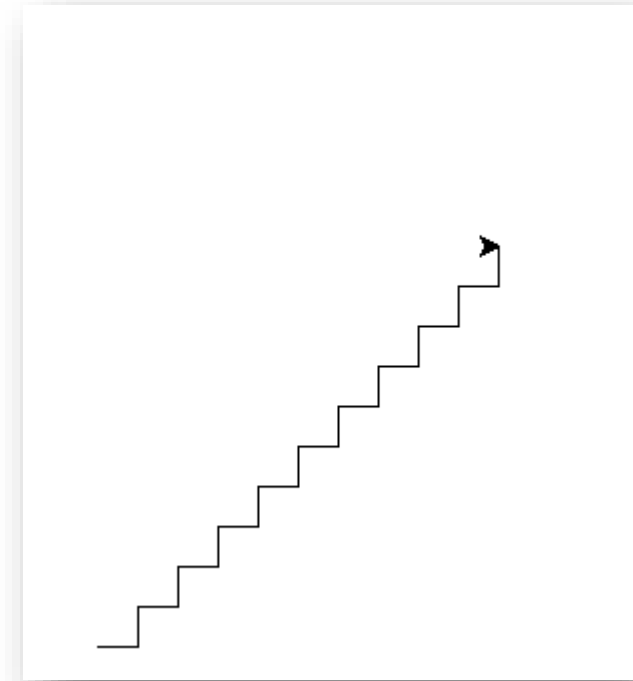
Merdiven çizelim

```
merdiven.py - C:\Python34\merdiven.py (3.4.4)
File Edit Format Run Options Window Help
from turtle import *          #turtle modülü import et

def merdiven(basamak sayısı,uzunluk): #merdiven isiminde parametrelili fonksiyon tanımla
    speed(1)          #turtle hızını 1 yap( 1 yavaş-10 hızlı)
    penup()           #kalemi kaldır
    goto(-300,-300)   #penceredeki -300-, -300 konumuna git
    pendown()         #kalemi bastır
    for x in range(basamak sayısı): #basamak sayısı kadar tekrarla
        forward(uzunluk) #uzunluk kadar ileri çizgi çiz
        left(90)         #90 derece sola dön
        forward(uzunluk) # uzunluk kadar ileri çizgi çiz
        right(90)        #90 derece sağa dön

merdiven(5,100) # merdiven fonksiyonunu çağır basamak için 5 uzunluk için 100 değerini gönder
clear() #ekranı temizle
merdiven(10,50) # merdiven fonksiyonunu çağır basamak için 10 uzunluk için 50 değerini gönder
clear() #ekranı temizle
merdiven(10,20) # merdiven fonksiyonunu çağır basamak için 30 uzunluk için 20 değerini gönder

Ln: 19 Col: 11
```



TURTLE MODÜLÜ ÖĞRENDİĞİMİZ FONKSİYONLAR

Fonksiyon adı-kullanımı	Açıklaması
forward(100)	100 birim ileri çizgi çiz
backward(50)	50 birim geri çizgi çiz
left(60)	60 derece sola dön
right(90)	90 derece sağa dön
pensize(10)	Kalem ucu kalınlığını 10 birim yap
color("red","yellow")	Çizgi rengini kırmızı, dolgu rengini sarı yap
begin_fill()	Boyamayı başlat
end_fill()	Boyamayı bitir
circle(50)	50 birimlik daire çiz
speed(1)	turtle hızını ayarla(1 yavaş-10 hızlı)
penup()	kalemi kaldır
pendown()	kalemi bastır
goto(100,200)	pencere de x =100 ,y =200 koordinatına git
clear()	ekranı temizle

Turtle modülü için kaynak adresi

<https://docs.python.org/3.3/library/turtle.html?highlight=turtle>

TIME MODÜLÜ

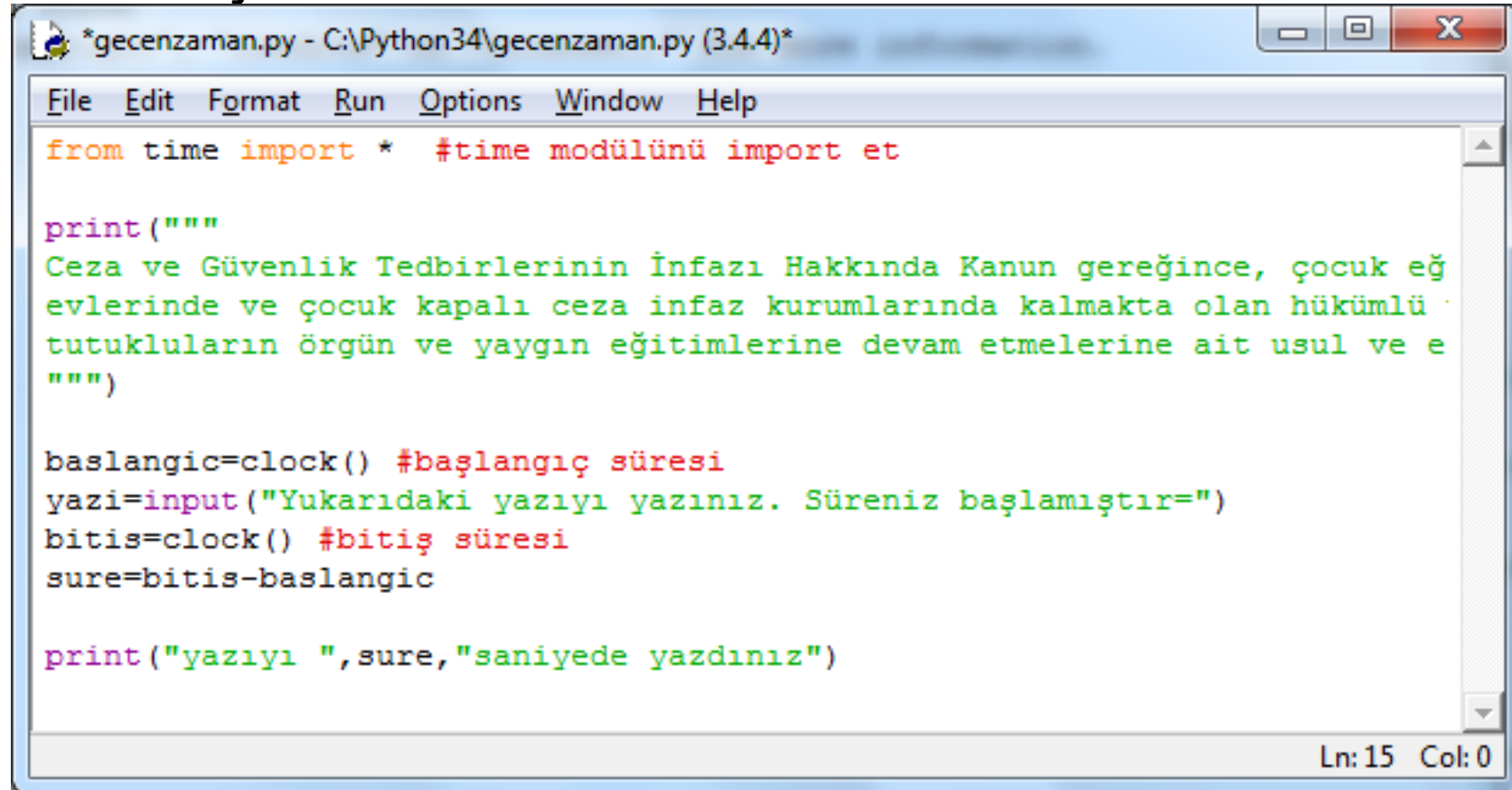
Time Modülü

Zamanla ilgili bilgi ve işlemlerin yer aldığı modül, “time” modülüdür. Time modülünü import etmek için aşağıdaki kod satırı yazılmalıdır

```
from time import *
```


clock() fonksiyonu

clock() fonksiyonu ile programın belli bölümlerinin çalışma süresini saniye cinsinden ölçebiliriz.



```
*gecenzaman.py - C:\Python34\gecenzaman.py (3.4.4)*
File Edit Format Run Options Window Help
from time import * #time modülünü import et

print("""
Ceza ve Güvenlik Tedbirlerinin İnfazı Hakkında Kanun gereğince, çocuk eğ
evlerinde ve çocuk kapalı ceza infaz kurumlarında kalmakta olan hükümlü
tutukluların örgün ve yaygın eğitimlerine devam etmelerine ait usul ve e
""")

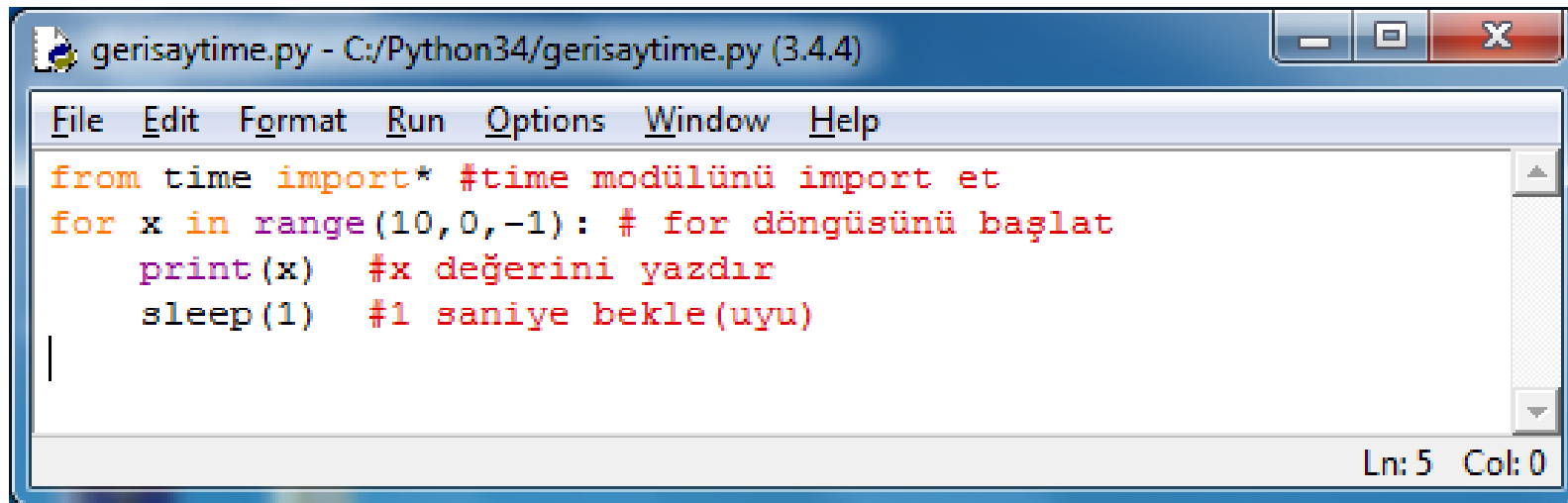
baslangic=clock() #başlangıç süresi
yazi=input("Yukarıdaki yazıyı yazınız. Süreniz başlamıştır=")
bitis=clock() #bitiş süresi
sure=bitis-baslangic

print("yazıyı ",sure,"saniyede yazdınız")

Ln: 15 Col: 0
```

sleep() Fonksiyonu

- Sleep() fonksiyonu ise programın çalışması sırasında belirtilen süre kadar durmasını sağlar.
- Örneğin geriye sayımda her sayıdan sonra 1 saniye beklemek için aşağıda görülen kod kullanılır.

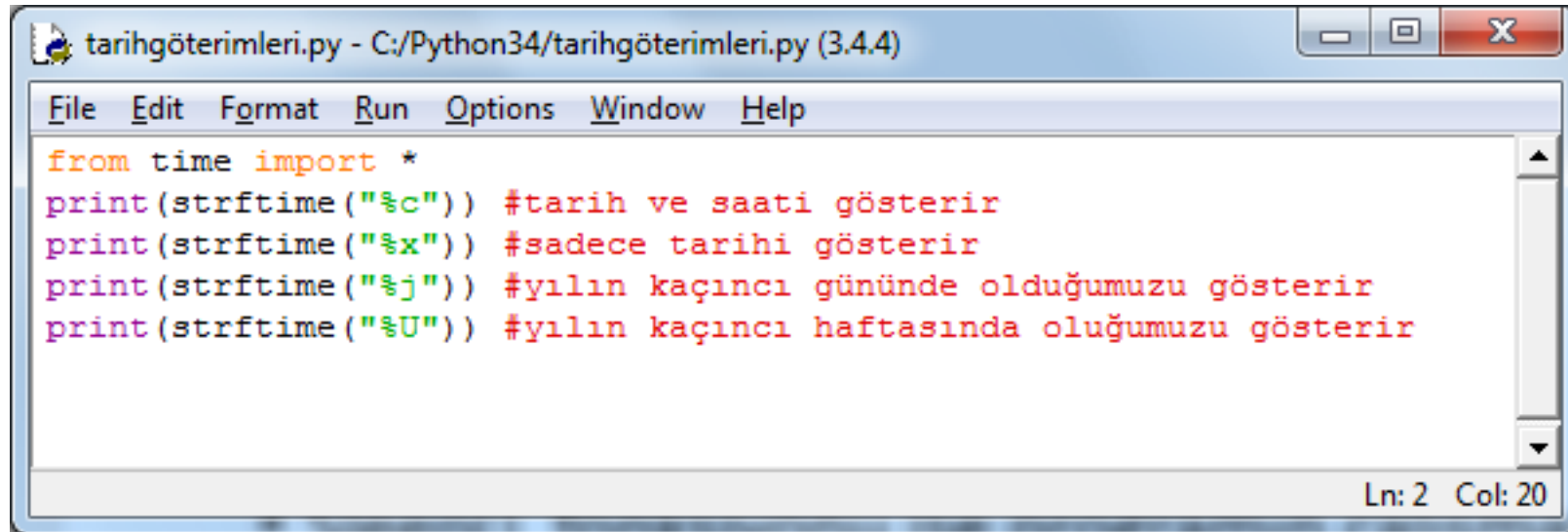


```
gerisaytime.py - C:/Python34/gerisaytime.py (3.4.4)
File Edit Format Run Options Window Help
from time import* #time modülünü import et
for x in range(10,0,-1): # for döngüsünü başlat
    print(x) #x değerini yazdır
    sleep(1) #1 saniye bekle(uyu)
|
Ln: 5 Col: 0
```

```
10
9
8
7
6
5
4
3
2
1
>>> |
```

strftime() fonksiyonu

Bu fonksiyon içerisine yazılan özel parametrelere göre zaman hakkında bilgi vermektedir.



```
tarihgöterimleri.py - C:/Python34/tarihgöterimleri.py (3.4.4)
File Edit Format Run Options Window Help
from time import *
print(strftime("%c")) #tarih ve saati gösterir
print(strftime("%x")) #sadece tarihi gösterir
print(strftime("%j")) #yılın kaçınıcı gününde olduğumuzu gösterir
print(strftime("%U")) #yılın kaçınıcı haftasında oluşumuzu gösterir
Ln: 2 Col: 20
```

```
04/27/18 14:33:40
04/27/18
117
16
>>>
```

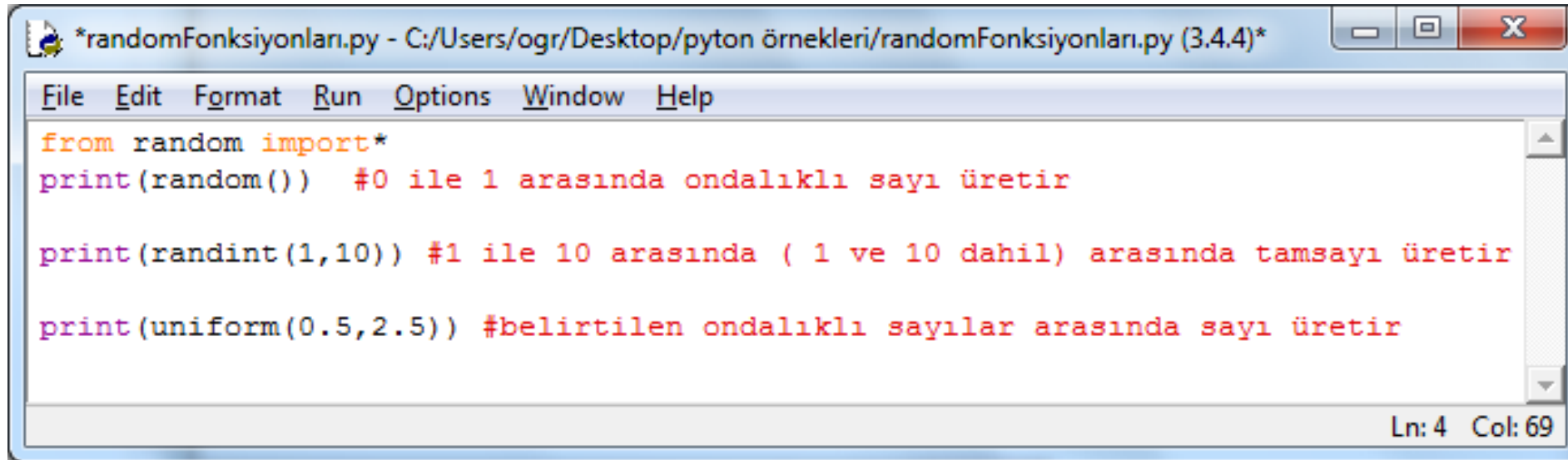
Random Modülü

Random Modülü Nedir

Bu modülü import ederek bilgisayarın rastgele sayılar üretmesini sağlayabiliriz. Random modülünü import etmek için aşağıdaki satır yazılmalıdır

```
from random import *
```

Random Fonksiyonlarının kullanımı



The screenshot shows a Python IDE window titled '*randomFonksiyonları.py - C:/Users/ogr/Desktop/pyton örnekleri/randomFonksiyonları.py (3.4.4)*'. The menu bar includes File, Edit, Format, Run, Options, Window, and Help. The code in the editor is as follows:

```
from random import*
print(random()) #0 ile 1 arasında ondalıklı sayı üretir

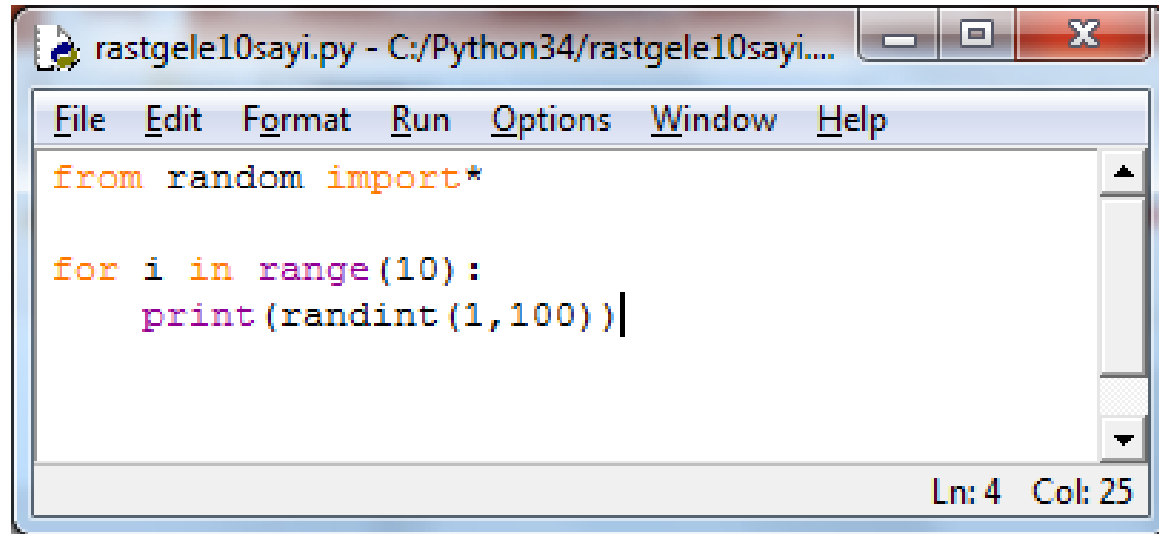
print(randint(1,10)) #1 ile 10 arasında ( 1 ve 10 dahil) arasında tamsayı üretir

print(uniform(0.5,2.5)) #belirtilen ondalıklı sayılar arasında sayı üretir
```

The status bar at the bottom right of the window indicates 'Ln: 4 Col: 69'.

```
0.5865020954771984
3
1.173638135720657
>>>
```

Rastegele 1 ile 100 arasında 10 adet tam sayı üreten kodları yazalım



```
File Edit Format Run Options Window Help
from random import*

for i in range(10):
    print(randint(1,100))|
```

Ln: 4 Col: 25

```
52
90
63
56
40
90
97
93
13
24
>>>
```

1-10 sayıları arasında bilgisayarın üreteceği sayıyı bulmaya çalışalım

```
sayı tahminrandom.py - C:\Python34\sayı tahminrandom.py (3.4.4)
File Edit Format Run Options Window Help
from random import*
sayı=randint(1,10)
for i in range(3):
    tahmin=int(input("1-10 arası bir sayı tahmin ediniz:"))
    if tahmin==sayı:
        print("tebrikler bildiniz")
        break
    else:
        print("bilemediniz")

print("bilgisayarın ürettiği sayı=",sayı)

Ln:1 Col:0
```

```
1-10 arası bir sayı tahmin ediniz:6
bilemediniz
1-10 arası bir sayı tahmin ediniz:3
tebrikler bildiniz
bilgisayarın ürettiği sayı= 3
>>> |
```


KENDİ MODÜLÜMÜZÜ OLUŞTURALIM

KENDİ MODÜLÜMÜZÜ OLUŞTURALIM

Kendi oluşturduğumuz fonksiyonları modül olarak kaydedebilir ve sayfalarımıza import edebiliriz.

- Bunun için modül içerisinde sadece fonksiyon tanımlaması yapılmalıdır.
- Modül Python>lib klasörüne kaydedilerek tüm çalışmalara import edilmelidir.

```
dortislem.py - C:\Python34\Lib\dortislem.py (3.4.4)
File Edit Format Run Options Window Help
#Modül içerisinde sadece fonksiyon tanımlaması yapılır

def topla(a,b):
    sonuc=a+b
    print("toplama işleminin sonucu=",sonuc)

def fark(a,b):
    sonuc=a-b
    print("Çıkarma işleminin sonucu=",sonuc)

def carp(a,b):
    sonuc=a*b
    print("Çarpma işleminin sonucu=",sonuc)

def bol(a,b):
    sonuc=a/b
    print("Çarpma işleminin sonucu=",sonuc)
```

```
dortislemcagir.py - C:/Python34/dortislem...
File Edit Format Run Options Window Help
from dortislem import *

topla(20,30)
fark(10,2)
carp(5,8)
bol(20,10)
|
Ln: 7 Col: 0
```

```
toplama işleminin sonucu= 50
Çıkarma işleminin sonucu= 8
çarpma işleminin sonucu= 40
Çarpma işleminin sonucu= 2.0
>>>
```

"dortislem" adındaki modülümüz ve içerisinde yer alan fonksiyonlar oluşturuldu ve Bu dosya "lib " klasörüne kaydedildi.

Önemli Modül içerisinde sadece fonksiyonlar olacak

Daha sonra modül import edildi ve fonksiyonlar çağırılarak gerekli sonuçlar alındı