

ANR 413 Bilgisayar Programlama

Python

#5

Serdar ARITAN

Biyomekanik Araştırma Grubu,
Spor Bilimleri Fakültesi
Hacettepe Üniversitesi
Ankara



Fonksiyonlar kısaca belirli bir işlevi yapmak için oluşturulan komutlar kümesidir. Örneğin `print()` fonksiyonu ekrana çıktı vermemizi sağlayan işlevi yapmak için belirli komutlar bulundurur.

Fonksiyon tanımlayarak yapmak istediğimiz işler için her seferinde kod yazmak yerine bir defa fonksiyon oluşturuyoruz o işlemi yapmak istediğimiz zaman o fonksiyonu çağırıp kullanıyoruz.

Kısaca Fonksiyonlar;

- Her biri belli bir işi yerine getiriyor.
- Fonksiyonların görevi, karmaşık işlemleri bir araya toplayarak, bu işlemleri tek adımda yapmamızı sağlamaktır.
- Fonksiyonları kullanarak, bir veya birkaç adımdan oluşan işlemleri tek bir isim altında toplayabiliriz.

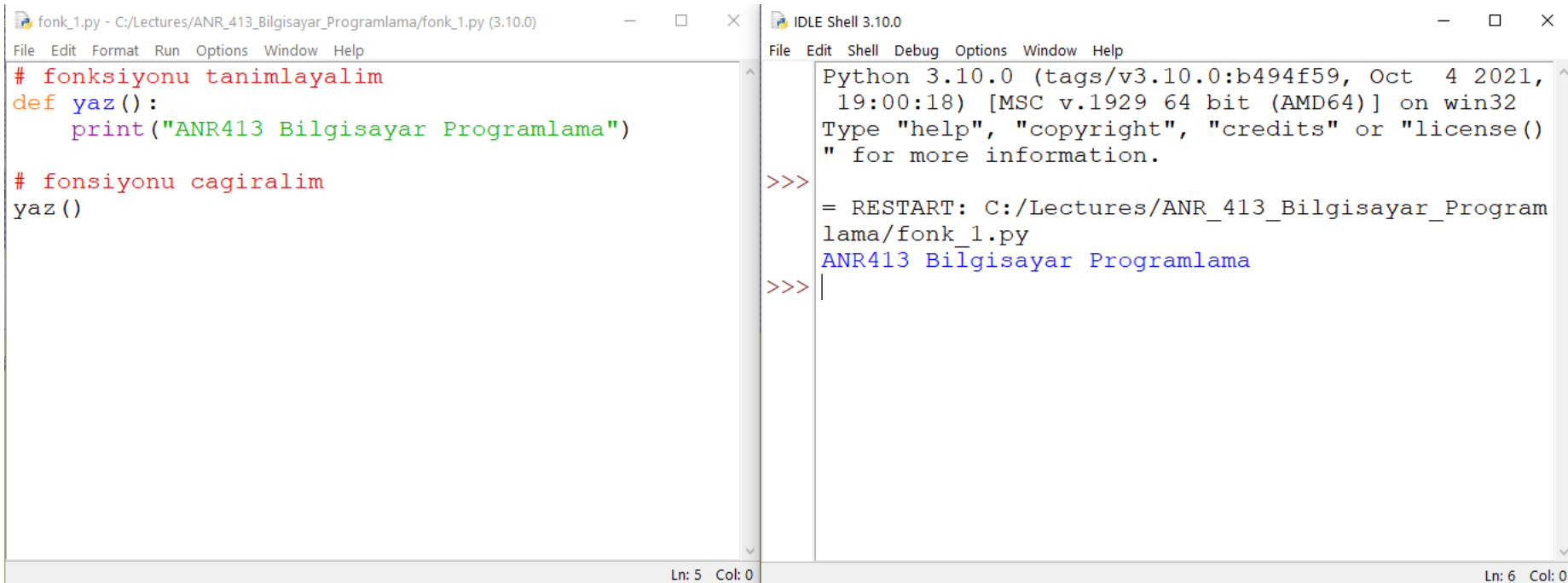
Python da iki tür fonksiyon vardır birincisi Python ile hazır gelen fonksiyonlar yani gömülü fonksiyonlar, ikincisi ise bizim yazdığımız fonksiyonlar. `print()` ve `input()` gibi fonksiyonlar Python ile birlikte hazır geldiği için biz sadece onları kullanmak istediğimiz durumda çağırıyoruz.

Fonksiyonların genel kullanım taslağı şu şekildedir.

```
>>> def fonksiyon_Adi(parametre1,parametre2):
```

```
    yapılacak işlemler
```

```
>>> fonksiyon_Adi(isim, no)
```



The image shows a screenshot of the Python IDLE Shell interface. The left pane displays a Python script named `fonk_1.py` with the following code:

```
# fonksiyonu tanımlayalım
def yaz():
    print("ANR413 Bilgisayar Programlama")

# fonsiyonu cagiralım
yaz()
```

The right pane shows the IDLE Shell output, which includes the Python version information and the execution of the `yaz()` function:

```
Python 3.10.0 (tags/v3.10.0:b494f59, Oct 4 2021,
19:00:18) [MSC v.1929 64 bit (AMD64)] on win32
Type "help", "copyright", "credits" or "license()"
>>>
= RESTART: C:/Lectures/ANR_413_Bilgisayar_Programlama/fonk_1.py
ANR413 Bilgisayar Programlama
>>> |
```

The status bar at the bottom indicates the current line and column numbers: `Ln: 5 Col: 0` for the script editor and `Ln: 6 Col: 0` for the shell.

```
fonk_1.py - C:/Lectures/ANR_413_Bilgisayar_Programlama/fonk_1.py (3.10.0)
File Edit Format Run Options Window Help
# fonksiyonu tanımlayalım
def yaz():
    print("ANR413 Bilgisayar Programlama")

# fonsiyonu cagiralim
yaz()
yaz()
yaz()

Ln: 9 Col: 0
```

```
IDLE Shell 3.10.0
File Edit Shell Debug Options Window Help
Python 3.10.0 (tags/v3.10.0:b494f59, Oct 4 2021, 19:00:18) [MSC v.1929 64 bit (AMD64)] on win32
Type "help", "copyright", "credits" or "license()" for more information.

>>>
= RESTART: C:/Lectures/ANR_413_Bilgisayar_Programlama/fonk_1.py
ANR413 Bilgisayar Programlama

>>>
= RESTART: C:/Lectures/ANR_413_Bilgisayar_Programlama/fonk_1.py
ANR413 Bilgisayar Programlama
ANR413 Bilgisayar Programlama
ANR413 Bilgisayar Programlama

>>>

Ln: 11 Col: 0
```

```
fonk_2.py - C:/Lectures/ANR_413_Bilgisayar_Programlama/fonk_2.py (3.10.0)
File Edit Format Run Options Window Help
# fonksiyonlari tanimlayalim
def yaz():
    print("ANR413 Bilgisayar Programlama")

def selamla(isim):
    print("Merhaba" + isim)

# fonsiyonlari cagiralim
yaz()
selamla('Serdar')
yaz()
```

```
IDLE Shell 3.10.0
File Edit Shell Debug Options Window Help
Type "help", "copyright", "credits" or "license()" for more information.
>>>
= RESTART: C:/Lectures/ANR_413_Bilgisayar_Programlama/fonk_1.py
ANR413 Bilgisayar Programlama
>>>
= RESTART: C:/Lectures/ANR_413_Bilgisayar_Programlama/fonk_1.py
ANR413 Bilgisayar Programlama
ANR413 Bilgisayar Programlama
ANR413 Bilgisayar Programlama
>>>
= RESTART: C:/Lectures/ANR_413_Bilgisayar_Programlama/fonk_2.py
ANR413 Bilgisayar Programlama
MerhabaSerdar
ANR413 Bilgisayar Programlama
>>>
```

Ln: 5 Col: 18

Ln: 16 Col: 0



Fonksiyon Örneği

[illegible]



Fonksiyon Örneği

The screenshot displays two side-by-side windows from the Python IDLE 3.10.0 environment.

The left window, titled "fonk_3.py - C:/Lectures/ANR_413_Bilgisayar_Programlama/fonk_3.py (3.10.0)", contains the following Python code:

```
# fonksiyonlari tanimlayalim  
  
def sayi_topla(sayi1, sayi2):  
    return sayi1 + sayi2  
  
s1 = int(input('Bir sayi giriniz'))  
s2 = int(input('Bir daha sayi giriniz'))  
print('Iki sayinin toplami : ', sayi_topla(s1,s2))
```

The right window, titled "IDLE Shell 3.10.0", shows the output of running the script. It features a menu bar (File, Edit, Shell, Debug, Options, Window, Help) and a command prompt interface where the user has entered three restart commands followed by inputs:

```
>>>  
>>>  
>>>  
>>>  
>>>  
>>>  
>>>  
>>>  
>>>  
  
= RESTART: C:/Lectures/ANR_413_Bilgisayar_Program  
lama/fonk_3.py  
Iki sayinin toplami : 15  
  
>>>  
  
= RESTART: C:/Lectures/ANR_413_Bilgisayar_Program  
lama/fonk_3.py  
Bir sayi giriniz3  
Bir daha sayi giriniz5  
Iki sayinin toplami : 8  
  
>>> |
```

```
def toplama(s1,s2):  
    return s1+s2  
  
def cikarma(s1,s2):  
    return s1-s2  
  
def carpma(s1,s2):  
    return sayi1*sayi2  
  
def bolme(s1,s2):  
    return s1/s2
```

```
while True:  
    print("""\ttoplama icin 1  
            \tcikarma icin 2  
            \tcarpma icin 3  
            \tbolme icin 4  
            \tsecimden cikmak için 'q' harfine basiniz """)  
  
    secim = input("Bir Seçim Yapın :")  
  
    if secim == "q":  
        break  
  
    sayi1 = int(input("Birinci sayıyı girin :"))  
    sayi2 = int(input("İkinci sayıyı girin :"))  
  
    if secim == "1":  
        print("Sonuç :", toplama(sayi1, sayi2))  
    elif secim == "2":  
        print("Sonuç :", cikarma(sayi1,sayi2))  
    elif secim == "3":  
        print("Sonuc :", carpma(sayi1,sayi2))  
    elif secim == "4":  
        print("Sonuç :", bolme(sayi1,sayi2))  
    else:  
        print("Yanlış seçim lütfen tekrar deneyin")
```

```
        toplama icin 1
        cikarma icin 2
        carpma icin 3
        bolme icin 4
        secimden cikmak için 'q' harfine basiniz
Bir Seçim Yapın :1
Birinci sayıyı girin :12
İkinci sayıyı girin :5
Sonuç : 17
        toplama icin 1
        cikarma icin 2
        carpma icin 3
        bolme icin 4
        secimden cikmak için 'q' harfine basiniz
Bir Seçim Yapın :3
Birinci sayıyı girin :8
İkinci sayıyı girin :3
Sonuc : 24
        toplama icin 1
        cikarma icin 2
        carpma icin 3
        bolme icin 4
        secimden cikmak için 'q' harfine basiniz
Bir Seçim Yapın :q
```

```
toplama icin 1
cikarma icin 2
carpma icin 3
bolme icin 4
secimden cikmak için 'q' harfine basiniz
```

```
Bir Seçim Yapın :2
Birinci sayıyı girin :5
İkinci sayıyı girin :2
Sonuç : 3
```

```
toplama icin 1
cikarma icin 2
carpma icin 3
bolme icin 4
secimden cikmak için 'q' harfine basiniz
```

 5!

```
Bir Seçim Yapın :5
Birinci sayıyı girin :4
İkinci sayıyı girin :5
```

```
Yanlış seçim lütfen tekrar deneyin
```

```
toplama icin 1
cikarma icin 2
carpma icin 3
bolme icin 4
secimden cikmak için 'q' harfine basiniz
```

```
Bir Seçim Yapın :q
```



```
toplama icin 1
cikarma icin 2
carpma icin 3
bolme icin 4
secimden cikmak için 'q' harfine basiniz
Bir Seçim Yapın :2
Birinci sayıyı girin :5
İkinci sayıyı girin :2
Sonuç : 3
```

```
toplama icin 1
cikarma icin 2
carpma icin 3
bolme icin 4
secimden cikmak için 'q' harfine basiniz
```

5

```
Bir Seçim Yapın :5
Yanlış seçim lütfen tekrar deneyin
```

```
toplama icin 1
cikarma icin 2
carpma icin 3
bolme icin 4
secimden cikmak için 'q' harfine basiniz
```

```
Bir Seçim Yapın :q
```



```
def kutuCiz():  
    print("*****")  
    print("*           *")  
    print("*           *")  
    print("*****")
```

kutuCiz fonksiyonu doğru çalışıyor ancak sadece belirli bir kutuyu çiziyor. Esnekliği olmayan bir fonksiyon.



```
def kutuCiz(en, boy):  
    # Boyutları 2x2 den küçük kutular cizilemez  
  
    if en < 2 or boy < 2:  
        print("Hata: En veya boy çok küçük.")  
        quit()  
    # Kutunun ustunu ciz  
    print("*" * en)  
    # Kutunun kenarlarını ciz  
    for i in range(boy - 2):  
        print("*" + " " * (en - 2) + "*")  
    # Kutunun altını ciz  
    print("*" * en)  
  
kutuCiz(15, 4)
```

- `kutuCiz` fonsiyonunu 4 parametre alacak şekilde yeniden yazınız.

```
def kutuCiz(en, boy, dis ="*", ic =" "):
```

```
kutuCiz(14, 5, "@", ".")
```

```
@@@@@@@@@@@@@@
@.....@
@.....@
@.....@
@@@@@@@@@@@@@
```

- `vkiHesapla` adında bir fonsiyon yazınız. Bu fonsiyon vucut ağırlığını ve boy uzunluğunu parametre olarak alacak.

```
def vkiHesapla(va, boy):
```

- `vkiSonuc` adında bir fonsiyon yazınız. Bu fonsiyon hesaplanan vucut kitle indeksi (vki) değerini parametre olarak alacak.

```
def vkiSonuc(vki):
```