
Python ile Programlamaya Giriş

DERS 10: FONKSİYONLAR (DEVAM)

DR. HÜSEYİN BAHTİYAR



Değişken Uzunluklu Argümanlar

- ❖ Özel durumlarda fonksiyonun bizim tanımladığımızdan daha fazla argüman almasını isteyebiliriz.
- ❖ Bu argümanlara **değişken uzunluklu argümanlar** adı verilir.
- ❖ Fonksiyon tanımını içinde özel olarak tanımlanmamışlardır. Değişken uzunluklu argüman tanımlamak istiyorsak önüne * işareti koyarız.

Değişken Uzunluklu Argümanlar

```
def fonksiyon_adi([normal_degisken,] *degisken_uzunluklu_arguman ):
    "fonksiyon islemleri"
    function_suite
    return [donus_degeri]
```

```
#!/usr/bin/python

# Function definition is here
def printinfo( arg1, *vartuple ):
    "This prints a variable passed arguments"
    print "Output is: "
    print arg1
    for var in vartuple:
        print var
    return;

# Now you can call printinfo function
printinfo( 10 )
printinfo( 70, 60, 50 )
```

Örnek: Celcius-> Fahrenheit

- ▶ 3. Derste gördüğümüz örneği fonksiyon ile yazmaya çalışalım?
- ▶ Santigrad derece cinsinden verilmiş bir sıcaklığı Fahrenheit birimine çevirmek istediğimizde bu sayıyı 1.8 ile çarptıktan sonra 32 eklememiz gerekmektedir.
- ▶ **Formülümüz:**
- ▶ $F = C * 1.8 + 32.00$
- ▶ Python'da klavyeden girilen bir sıcaklığı fahrenheit cinsine bir fonksiyon ile çeviren program yazınız.

Örnek

```
#!/python3
|
def ctoF(ce1):
    Fah=ce1*1.8 +32
    return Fah

celcius= input("Sayi giriniz: ")
celcius=float(celcius)
print("Cel: ",celcius," Fah: ",ctoF(celcius))
```

Örnek Devam

- ❖ Şimdi ise elimizde bir liste olsun bu listeyi fonksiyona gönderelim.
- ❖ Bize fahrenheit sonuçlarını liste olarak döndürsün

l=[1,43,54,12]

Örnek Devam

```
#!/python3

def ctoF(ce1):
    Fah=list()
    for var in ce1:
        Fah.append(var*1.8 +32)
    return Fah
l=[1,43,54,12]
print(" Fah: ",ctoF(l))
```

Örnek Değişken Uzunluk

- ❖ Son olarak, az önce öğrendiğimiz gibi değişken uzunluklu yapıyı kullanarak aynı fonksiyonu yazalım.

Örnek Değişken Uzunluk

```
#!/python3

def ctoF(*cel):
    Fah=list()
    for var in cel:
        Fah.append(var*1.8 +32)
    return Fah
print(" Fah: ",ctoF(1,43,54,12))
```

Fark Nedir?

- ❖ Bu iki fonksiyonun farkı nedir?

```
#!/python3

def ctoF(cel):
    Fah=list()
    for var in cel:
        Fah.append(var*1.8 +32)
    return Fah
l=[1,43,54,12]
print(" Fah: ",ctoF(l))
```

```
#!/python3

def ctoF(*cel):
    Fah=list()
    for var in cel:
        Fah.append(var*1.8 +32)
    return Fah
print(" Fah: ",ctoF(1,43,54,12))
```

Anonim Fonksiyonlar

- ❖ Bu fonksiyonları standard yapıda (def kullanarak) tanımlamadığımız için ismi anonim fonksiyonlardır.
- ❖ **lambda** anahtar kelimesini kullanarak küçük bir anonim fonksiyon oluşturabiliriz.
 - ❖ lambda'nın alacağı parametreler için bir sınırlama yoktur fakat sadece tek bir değer döndürür.
 - ❖ Komut veya birden fazla ifade içeremez.
 - ❖ Anonim fonksiyonun sonucunu doğrudan print içinde kullanamayız.
 - ❖ Anonim fonksiyonların kendi yerel namespace i vardır bu sebeple kendi parametreleri ve global namespacedeki değişkenler dışındaki değişkenlere erişemezler.

Anonim Fonksiyonlar

- ❖ Anonim fonksiyon yapısını kullanarak toplama fonksiyonu yazalım.

```
# Function definition is here
sum = lambda arg1, arg2: arg1 + arg2;

# Now you can call sum as a function
print "Value of total : ", sum( 10, 20 )
print "Value of total : ", sum( 20, 20 )
```

Anonim Fonksiyonlar

- ❖ Celcius->Fahrenheit fonksiyonunu lambda ile yazalım?

```
#!/python3  
  
ctoF = lambda arg1: arg1*1.8 + 32;  
print(" Fah: ",ctoF(23))
```

Acknowledgements / Contributions

These slides are Copyright 2010- Charles R. Severance (www.dr-chuck.com) of the University of Michigan School of Information and open.umich.edu and made available under a Creative Commons Attribution 4.0 License. Please maintain this last slide in all copies of the document to comply with the attribution requirements of the license. If you make a change, feel free to add your name and organization to the list of contributors on this page as you republish the materials.

Initial Development: Charles Severance, University of Michigan School of Information

... Insert new Contributors and Translators here

