

Lambda İfadeleri

Tek satırlık kısa fonksiyonlar tanımlamak için Lambda ifadeleri kullanılır. Lambda ifadeleri anonim (isimsiz) fonksiyonlardır. Kullanım şekli:

```
Sonuc = (lambda değişkenler: işlemler) (parametreler)
```

Örneğin iki sayının toplamını hem klasik fonksiyon tanımlaması ile hemde lambda ifadeleri aşağıdaki gibi kodlayabiliriz:

İki sayı toplamı: def fonksiyonu ile	Eşdeğeri: Lambda fonksiyonu ile
<pre>def toplama(a, b): return a+b #Ana program print(toplama(3,5)) # 8</pre>	<pre>sonuc = lambda a, b: a+b #Ana program print(sonuc(3,5)) # Çıktı: 8</pre>

Hatta yukarıdaki eşdeğer lambda ifadesini aşağıdaki gibi tek satırda ifade edebiliriz;

```
print ((lambda a, b: a+b) (3,5)) # Çıktı: 8
```

Örnek 1.

Ana programda (main fonksiyonunda) girilen Fahrenheit sıcaklığını (F°) **cevir** isimli fonksiyonla Dereceye (C°) çeviren ve çevrim sonucunu ana programda gösteren programı yazınız. { Çevrim için $C^{\circ} = (F^{\circ} - 32) / 1.8$ formülünü kullanınız }.

Çözüm-1:

Python Dili Kodlaması	Ekran Çıktısı
<pre>#cevir fonksiyonu def cevir(f): return ((f - 32) / 1.8) #Ana program f = float(input('Fahrenheit değeri:')) C = cevir(f) print(C , " derecedir")</pre>	<pre>Fahrenheit değeri:68 20.0 derecedir</pre>

Şimdi bu örnekte verilen Fahrenheit sıcaklığını (F°) Dereceye (C°) çeviren programı **lambda** ifadeleri ile kodlayalım. **Çözüm-2.**

Python Dili Kodlaması	Ekran Çıktısı
<pre>print ((lambda f: (f - 32) / 1.8) (212))</pre>	<pre>100.0</pre>

Örnek 2.

Şimdi girilen bir sayının **faktöriyelini alan** programı kodlayalım.

Çözüm.

Lambda ile kodlanması	Özyinelemeli (rikürsif) yaklaşım
<pre>fakt = lambda n: 1 if n <= 1 else n * fakt(n - 1)</pre> <pre>#Ana program n = int (input("Faktöriyeli alınacak sayı:")) print (fakt(n))</pre>	<pre>def fakt(n): if n <= 1: return 1 else: return n *fakt(n-1)</pre> <pre>#return 1 if n <= 1 else n * fakt(n - 1)</pre> <pre>#Ana program n = int (input("Faktöriyeli alınacak sayı:")) print (fakt(n))</pre>

*Lambda ifadeleri ile **map()**, **filter()** ve **reduce()** fonksiyonları sıklıkla kullanılır. Bu fonksiyonların lambda ile kullanımı için ;*

reduce() fonksiyonu

Bir grup veriden (listeden) tek bir anlamlı değer (sonuç) üretmek için **reduce(azaltma/indirgeme)** fonksiyonu kullanılır. **reduce()** fonksiyonu '**functools**' paketine ihtiyaç duyduğu için öncesinde bu paketin import edilmesi gerekir.

Örnek 3.

1'den 100'e kadar sayıların toplamını **lambda ve reduce() fonksiyonu** ile kodlayalım.

Çözüm.

Python Dili Kodlaması	Ekran Çıktısı
<pre>from functools import reduce liste = list(range(1,101)) #1-100 arası sayıları tutan liste toplam = reduce(lambda x, y: x+y, liste) print (toplam)</pre>	5050

filter() fonksiyonu

Bir grup veriyi (listeyi) belli bir kritere (veya fonksiyona) göre filtrelendirmek için **filter()** fonksiyonu kullanılır. Filter fonksiyonu ile mevcut liste içerisinde istenilen şarta uygun yeni listeler oluşturulabilir.

Örnek 4.

1...100 arasındaki tam sayılardan oluşan bir listeden **filter()** fonksiyonu ile tek sayıları filtreleyip yeni liste oluşturulalım.

Çözüm.

Python Dili Kodlaması	Ekran Çıktısı
<pre>liste = list(range(1,101)) #1-100 arası sayıları tutan liste tekS= list(filter(lambda x: x%2!=0, liste)) print (tekS)</pre>	<pre>[1, 3, 5, 7, 9, 11, 13, 15, 17, 19, 21, 23, 25, 27, 29, 31, 33, 35, 37, 39, 41, 43, 45, 47, 49, 51, 53, 55, 57, 59, 61, 63, 65, 67, 69, 71, 73, 75, 77, 79, 81, 83, 85, 87, 89, 91, 93, 95, 97, 99]</pre>

map() fonksiyonu

Bir grup veriyi (liste, demet) belli bir kriter (veya fonksiyon) ile eşleştirmek (haritalandırmak) için **map()** fonksiyonu kullanılır. Map fonksiyonu ile mevcut listeden yeni listeler oluşturulur. Örneğin bir string listenin karakter uzunluklarından oluşan yeni bir liste aşağıdaki gibi tek bir satırda oluşturulabilir;

❖ `list(map(len, ['abc', 'de', 'fghi']))` #Çıktısı:[3, 2, 4]

Örnek 5.

Bir listedeki sayıların karesinden yeni bir liste oluşturan programı **map()** fonksiyonu ile kodlayalım.

Çözüm.

Python Dili Kodlaması	Ekran Çıktısı
<pre>liste = [1, 2, 3, 4, 5] kare = map(lambda x: x **2, liste) print(list(kare))</pre>	<pre>[1, 4, 9, 16, 25]</pre>