

# Python Alıştırmalar

# Uygulama: Python görevlerini tamamlayınız.

## Görev 1: Verilen değerlerin veri yapılarını inceleyiniz.

```
x = 8

y = 3.2

z = 8j + 18

a = "Hello World"

b = True

c = 23 < 22

l = [1, 2, 3, 4]

d = {"Name": "Jake",
     "Age": 27,
     "Adress": "Downtown"}

t = ("Machine Learning", "Data Science")

s = {"Python", "Machine Learning", "Data Science"}
```



Type() metodunu kullanınız.

**Görev 2:** Verilen string ifadenin tüm harflerini büyük harfe çeviriniz. Virgül ve nokta yerine space koyunuz, kelime kelime ayırınız.

```
text = "The goal is to turn data into information, and information into insight."
```

**Beklenen çıktı:**

```
['THE', 'GOAL', 'IS', 'TO', 'TURN', 'DATA', 'INTO', 'INFORMATION', 'AND', 'INFORMATION', 'INTO', 'INSIGHT']
```



String metodlarını kullanınız.

**Görev 3:** Verilen listeye aşağıdaki adımları uygulayınız.

```
lst = ["D", "A", "T", "A", "S", "C", "I", "E", "N", "C", "E"]
```

**Adım 1:** Verilen listenin eleman sayısına bakınız.

**Adım 2:** Sıfırinci ve onuncu indeksteki elemanları çağırınız.

**Adım 3:** Verilen liste üzerinden ["D", "A", "T", "A"] listesi oluşturunuz.

**Adım 4:** Sekizinci indeksteki elemanı siliniz.

**Adım 5:** Yeni bir eleman ekleyiniz.

**Adım 6:** Sekizinci indekse "N" elemanını tekrar ekleyiniz.

**Görev 4:** Verilen sözlük yapısına aşağıdaki adımları uygulayınız.

```
dict = {'Christian': ["America",18],  
        'Daisy':["England",12],  
        'Antonio':["Spain",22],  
        'Dante':["Italy",25]}
```

**Adım 1:** Key değerlerine erişiniz.

**Adım 2:** Value'lara erişiniz.

**Adım 3:** Daisy key'ine ait 12 değerini 13 olarak güncelleyiniz.

**Adım 4:** Key değeri Ahmet value değeri [Turkey,24] olan yeni bir değer ekleyiniz.

**Adım 5:** Antonio'yu dictionary'den siliniz.

**Görev 5:** Argüman olarak bir liste alan, listenin içerisindeki tek ve çift sayıları ayrı listelere atayan ve bu listeleri return eden fonksiyon yazınız.

```
l = [2,13,18,93,22]

def func(..):
    ...
    ...
    return ..

even_list, odd_list = func(l)
```



Liste elemanlarına tek tek erişmeniz gerekmektedir.

Her bir elemanın çift veya tek olma durumunu kontrol etmek için % yapısını kullanabilirsiniz.

**Görev 6:** List Comprehension yapısı kullanarak car\_crashes verisindeki numeric değişkenlerin isimlerini büyük harfe çeviriniz ve başına NUM ekleyiniz.

```
import seaborn as sns
df = sns.load_dataset("car_crashes")
df.columns
```

**Beklenen çıktı:**

```
[ 'NUM_TOTAL',
  'NUM_SPEEDING',
  'NUM_ALCOHOL',
  'NUM_NOT_DISTRACTED',
  'NUM_NO_PREVIOUS',
  'NUM_INS_PREMIUM',
  'NUM_INS_LOSSES',
  'ABBREV' ]
```



Numeric olmayan değişkenlerin de isimleri büyümeli.  
Tek bir list comprehension yapısı kullanılmalı.



**Görev 7:** List Comprehension yapısı kullanarak car\_crashes verisinde isminde "no" **barındırmayan** değişkenlerin isimlerinin **sonuna** "FLAG" yazınız.

**Beklenen çıktı:**

```
[ 'TOTAL_FLAG',  
  'SPEEDING_FLAG',  
  'ALCOHOL_FLAG',  
  'NOT_DISTRACTED',  
  'NO_PREVIOUS',  
  'INS_PREMIUM_FLAG',  
  'INS_LOSSES_FLAG',  
  'ABBREV_FLAG' ]
```



Tüm değişkenlerin isimleri büyük harf olmalı.  
Tek bir list comprehension yapısı ile yapılmalı.



**Görev 8:** List Comprehension yapısı kullanarak aşağıda verilen değişken isimlerinden FARKLI olan değişkenlerin isimlerini seçiniz ve yeni bir dataframe oluşturunuz.

```
og_list = ["abbrev", "no_previous"]
```

**Beklenen çıktı:**

total	speeding	alcohol	not_distracted	ins_premium	ins_losses
18.800	7.332	5.640	18.048	784.550	145.080
18.100	7.421	4.525	16.290	1053.480	133.930
18.600	6.510	5.208	15.624	899.470	110.350
22.400	4.032	5.824	21.056	827.340	142.390
12.000	4.200	3.360	10.920	878.410	165.630



Önce verilen listeye göre list comprehension kullanarak new\_cols adında yeni liste oluşturunuz.

Sonra df[new\_cols] ile bu değişkenleri seçerek yeni bir df oluşturunuz ve adını new\_df olarak isimlendiriniz.

**miuul**

miuul.com