

```
#yazı tura atarken, kaç deneyde kaç yazı gelebileceğine bakalım. Deney sayımız(n)
#10 olsun
import numpy as np
import collections
n, h_number=10,0 #n,deney sayısını; h_number, kaç defa tura geldiğini tutuyor.
t=[]
for i in range (n):
    outcome = np.random.choice(['H','T']) # H: tura, T:yazı
    t.append(outcome)
    if outcome =='H':
        h_number+=1
    collections.Counter(t)
print(h_number) #bu kod sadece n kadar deneyde kaç yazı geleceğine bakar.
#Oranı vermeyecektir
#fonksiyon haline getirip tekrar for döngüsüne alabiliriz.
```

```
#simdi bu olasılıkların birbirine yaklaşıp yaklaşmadığını görmek için
#fonksiyon yapalım
import numpy as np
import collections
def my_f(n=100):
    h_number=0
    t=[]
    for i in range (n):
        outcome = np.random.choice(['H','T']) # H: tura, T:yazı
        t.append(outcome)
        if outcome =='H':
            h_number+=1
        collections.Counter(t)
    return h_number/n
my_f() #parantez içine hangi sayı konursa deney o kadar tekrarlanır.
```

```
collections.Counter(t) #t listesine atılan yazı(T) ve tura(H) sayısını tutuyor.
```

```
#bu fonksiyonda da n defa zar atıp, belirlenen rakamın kaç kere gelebileceği
#olasılığına bakıyor.
def my_f_for_die(n=100000):#n'i yükselttikçe sonucun/sonuçların 0,16 çevresinde
#değiştiğini görüyoruz
    number_of_1=0
    number_of_4=0
    t=[]
    for i in range(n):
        outcome =np.random.choice([1,2,3,4,5,6])
        t.append(outcome)
        if outcome ==1:
            number_of_1 +=1
        if outcome ==4:
            number_of_4 +=1
    return number_of_1 / n , number_of_4 /n
my_f_for_die()
```

```
#simdi de üç kart bulunan bir torbadan bir kart seçip görünen yüzün rengine göre
#arkadaki rengi tahmin etmeue çalışalım
#üç karttan birinin iki yüzü de siyah, birinin siyah beyaz, birinin de iki yüzü
#de beyaz olsun.
#masaya konan kartın görünen yüzü siyah ise görünmeyen yüzün siyah olma olasılığı?
import random
c_1=['black', 'black']
c_2=['white', 'white']
c_3=['black', 'white']
cards=[c_1, c_2, c_3] #oluşturduğumuz kartları listeye aldık.Random bir seçim
#yapabilmek için

select_card=random.choice(cards)
print(select_card, end=" ")
select_face=random.choice(select_card)
select_card.remove(select_face)
unseen_face=select_card[0]
print("bilinen yüz:", select_face, ", bilinmeyen yüz:" ,unseen_face, end=" ")
```

```
#simdi bu kodu fonksiona çevirip birkaç olasılığı aynı anda görüp düşünelim
def my_f_5():
    b_number=0 #siyah gelen yüz sayısı başta 0 olur.
    c_1=['black', 'black']
    c_2=['white', 'white']
    c_3=['black', 'white']
    cards=[c_1, c_2, c_3]

    select_card=random.choice(cards)
    print(select_card, end=" ")

    select_face=random.choice(select_card)
    select_card.remove(select_face)
    unseen_face=select_card[0]
    print("bilinen:", select_face, ", bilinmeyen:" ,unseen_face, end=" ")

    if select_face=='black':
        b_number+=1 #süste gelen yüz siyahsa siyah yüz sayısını artırıyorum
        print(" +")#eğer üste gelen yüz siyahsa + veriyoruz.
    else:
        print()
    return b_number
#bu fonksiyon sadece tek bir değer döndürür
```

```
for i in range(10):
    my_f_5()
```

```
n=10
b=0
for i in range(n):
    b=my_f_5()
n,b #n,deney sayısı; b,siyah gelen yüz sayısını verecek
```

```
def my_f_5(p=False):
    b_number=0 #üste gelen siyah yüz sayısını tutacak
    w_number=0 #üste gelen siyah yüzün arka yüzünün beyaz olma durumunu
    #tutacak

    c_1=['black', 'black']
    c_2=['white', 'white']
    c_3=['black', 'white']
    cards=[c_1, c_2,c_3]

    select_card=random.choice(cards)

    if p:
        print(select_card, end=" ")
    select_face=random.choice(select_card)
    select_card.remove(select_face)
    unseen_face=select_card[0]
    if p:
        print(select_face, " bilinmeyen: ",unseen_face)
    if select_face=='black':
        b_number=b_number+1 #üste gelen yüz siyahsa, değişkeni artırdık
        if p:
            print("      +")
        if unseen_face=='white':
            w_number+=1 #arkadaki görünmeyen yüz beyazsa, değişkeni artırdık
    else:
        if p:
            print()
    return b_number, w_number
```

my_f_5()

```
def my_f_6(n=10):
    b=0 #5. fonksiyondan gelecek siyah yüz sayısını tutmak için yeni
    #bir değişken oluşturduk
    w=0 #5. fonksiyondan gelecek beyaz yüz sayısını tutmak için yeni
    #bir değişken oluşturduk. Gelecek rakamlarla toplanacak.
    for i in range(n):
        b=b+my_f_5()[0] #5. fonksiyondan gelen ilk indexi topladık
        w=w+my_f_5()[1] #5. fonksiyondan gelen 2. indexi topladık.
    return n,b,w #n,deney sayısı; b,siyah yüz sayısı, w beyaz yüz sayısı

my_f_6()
```

```
#simdi de n defa yapılan deneyde b kere üste gelen yüzün görünmeyen yüzünde
#de beyaz olma durumunu incelediğimizde çıkan oranları toplu bir şekilde
#inceleyebileceğimiz bir fonksiyon oluşturalım
def my_f_10(n=100):
    seen_face_number=0
    unseen_face_number=0

    for i in range(n):
        a,b,c=my_f_6(n) #6. fonksiyon üçlü dönüş yaptığı için üç değişkene
        #atadık.

        seen_face_number+=b #fonksiyondan gelecek üst yüz siyah sayısı(b)
        unseen_face_number+=c #fonksiyondan gelecek siyah yüzün arka beyaz
        #yüz sayısı(c)
    return int(seen_face_number/n), int(unseen_face_number/n)

for i in range(10):
    print(my_f_10())
```

my_f_10() #yani bu fonksiyon (üste gelen siyah yüz)/n(deneysayısı) oranını ve
#(üste gelen siyah yüzün arkasındaki yüzün beyaz olması)/n(deneysayısı) oranını
veriyor