

List Comprehension ve Pandas Ağıştırmalar

List Comprehension

Görev 1: 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 2: 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.


List Comprehension

Görev 3: 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.

Pandas Alıřtırmalar

Görev 1: Seaborn kütüphanesi içerisinde **Titanic** veri setini tanımlayınız.

Görev 2: Titanic veri setindeki kadın ve erkek yolcuların sayısını bulunuz.

Görev 3: Her bir sütuna ait unique değerlerin sayısını bulunuz.

Görev 4: pclass değişkeninin unique değerlerinin sayısını bulunuz.

Görev 5: pclass ve parch değişkenlerinin unique değerlerinin sayısını bulunuz.

Görev 6: embarked değişkeninin tipini kontrol ediniz. Tipini category olarak değiştiriniz ve tekrar kontrol ediniz.

Görev 7: embarked değeri C olanların tüm bilgilerini gösteriniz.

Görev 8: embarked değeri S olmayanların tüm bilgilerini gösteriniz.

Görev 9: Yaşı 30 dan küçük ve kadın olan yolcuların tüm bilgilerini gösteriniz.

Görev 10: Fare'i 500'den büyük veya yaşı 70 den büyük yolcuların bilgilerini gösteriniz.

Görev 11: Her bir değişkendeki boş değerlerin toplamını bulunuz.

Görev 12: who değişkenini dataframe'den çıkarınız.

Görev 13: deck değişkenindeki boş değerleri deck değişkenin en çok tekrar eden değeri (mode) ile doldurunuz.

Görev 14: age değişkenindeki boş değerleri age değişkenin medyanı ile doldurunuz.

Görev 15: survived değişkeninin pclass ve cinsiyet değişkenleri kırılımında sum, count, mean değerlerini bulunuz.

Görev 16: 30 yaşın altında olanlar 1, 30'a eşit ve üstünde olanlara 0 vericek bir fonksiyon yazın. Yazdığınız fonksiyonu kullanarak titanic veri setinde age_flag adında bir değişken oluşturunuz oluşturunuz. (apply ve lambda yapılarını kullanınız)

Görev 17: Seaborn kütüphanesi içerisinden **Tips** veri setini tanımlayınız.

Görev 18: Time değişkeninin kategorilerine (Dinner, Lunch) göre total_bill değerlerinin toplamını, min, max ve ortalamasını bulunuz.

Görev 19: Günlere ve time göre total_bill değerlerinin toplamını, min, max ve ortalamasını bulunuz.

Görev 20: Lunch zamanına ve kadın müşterilere ait total_bill ve tip değerlerinin day'e göre toplamını, min, max ve ortalamasını bulunuz.

Görev 21: size'i 3'ten küçük, total_bill'i 10'dan büyük olan siparişlerin ortalaması nedir? (loc kullanınız)

Görev 22: total_bill_tip_sum adında yeni bir değişken oluşturunuz. Her bir müşterinin ödediği totalbill ve tip in toplamını versin.

Görev 23: total_bill_tip_sum değişkenine göre büyükten küçüğe sıralayınız ve ilk 30 kişiyi yeni bir dataframe'e atayınız.