**VERİ SETİNİ OKUTMA**  
df = pd.read\_csv("C:/Users/ASUSVivaBook/Desktop/GitHub/Data\_Analytics/07 EDA/Midterm/starbucks.csv")

**VERİ SETİNİN İLK 5 SATIR VE SÜTUNLARI**   
df.head() -> en az 5 gösterir

**SÜTUNLARA BAKILIR**  
df.columns  
df.keys()

**VERİ SETİNİN GENEL BİLGİLERİ (SÜTUN TÜRLERİ, EKSİK DEĞERLER)**  
df.info()

**SÜTUN ADLARINDAKİ BOŞLUKLARI KALDIRMAK İÇİN**  
df.rename(columns=lambda x: x.strip(), inplace=True)  
inplace=True parametresi değişikliklerin orijinal DataFrame üzerinde yapılmasını sağlar

**VERİNİN TEMEL İSTAİSTİKLERİ**  
df.describe().T  
.T -> sütunları satır, satırları sütun yapar.

**YALNIZCA OBJECT SÜTUNLARDAKİ TEMEL İSTATİSTİKLERİ DİKKATE ALIR**  
df.describe(include='object').T

**ORTALAMA DEĞERİ HESAPLAR**  
df.groupby('Beverage').mean(numeric\_only=True)

**EN YÜKSEK / EN DÜŞÜK**df.nlargest(5, 'Sugars (g)')  
df.nsmallest(‘’)

**SADECE İSTEDİĞİMİZ SÜTUNLARI GETİRİYORUZ**  
max5 = df.nlargest(5, 'Sugars (g)')  
max5.index  
df.iloc[max5.index][['Beverage', 'Sugars (g)']]

**SAYISAL DEĞERE ÇEVİRME**  
df['Caffeine (mg)'] = pd.to\_numeric(df['Caffeine (mg)'], errors='coerce')

**GÖRSELLEŞTİRME**  
**1-** plt.bar(x = df.groupby('Beverage\_category')['Total Fat (g)'].mean().sort\_values(ascending=False).index,

height = df.groupby('Beverage\_category')['Total Fat (g)'].mean().sort\_values(ascending=False).values)

plt.xticks(rotation=90)

plt.show()

**2-** df.groupby('Beverage\_category')['Total Fat (g)'].mean().sort\_values(ascending=False).plot(kind='bar');

**3-** df.groupby('Beverage\_category')['Total Fat (g)'].mean().sort\_values(ascending=False)

sns.barplot(x = df.groupby('Beverage\_category')['Total Fat (g)'].mean().sort\_values(ascending=False).index,

y = df.groupby('Beverage\_category')['Total Fat (g)'].mean().sort\_values(ascending=False).values)

plt.xticks(rotation=90)

plt.show()

**HER BİR DEĞERİN SÜTUNDA KAÇ KEZ TEKRARLANDIĞINI GÖSTERİR**  
df['Vitamin A (% DV)'] .value\_counts()

**SÜTUNDAKİ DEĞERİ SAYISAL FORMATA ÇEVİRİR.**df['Vitamin C (% DV)'] = pd.to\_numeric(df['Vitamin C (% DV)'].str.rstrip('%'), errors='coerce').astype('float')  
df['Vitamin C (% DV)'].str.rstrip('%') -> sütundaki her bir değerin sonundaki % işaretini kaldırır.  
astype('float') -> sonuçtaki değeri ondalık sayı formatına çevirir.

**İÇECEK GRUPLARI ARASINDA EN FAZLA BOY FARKI OLAN İÇECEĞİ KARŞILAŞTIRMA**  
**1-** max\_calorie = df.loc[df['Calories'].idxmax()]

min\_calorie = df.loc[df['Calories'].idxmin()]

pd.concat([max\_calorie, min\_calorie], axis=1)

**2-** calorie\_by\_category\_max = df.groupby('Beverage\_category')['Calories'].max().sort\_index().to\_frame()

calorie\_by\_category\_max.rename(columns={'Calories':'Max\_Calorie'}, inplace=True)

calorie\_by\_category\_max

calorie\_by\_category\_min = df.groupby('Beverage\_category')['Calories'].min().sort\_index().to\_frame()

calorie\_by\_category\_min.rename(columns={'Calories':'Min\_Calorie'}, inplace=True)

calorie\_by\_category\_min

**YUKARIDAKİ İKİ SÜTUNU BİRLEŞTİRİRKEN** -> calorie\_by\_category = pd.merge(calorie\_by\_category\_max, calorie\_by\_category\_min, left\_index=True, right\_index=True)

calorie\_by\_category

**İKİ SÜTUNUN FARKI** -> calorie\_by\_category['Diff'] = calorie\_by\_category['Max\_Calorie'] - calorie\_by\_category['Min\_Calorie']

calorie\_by\_category