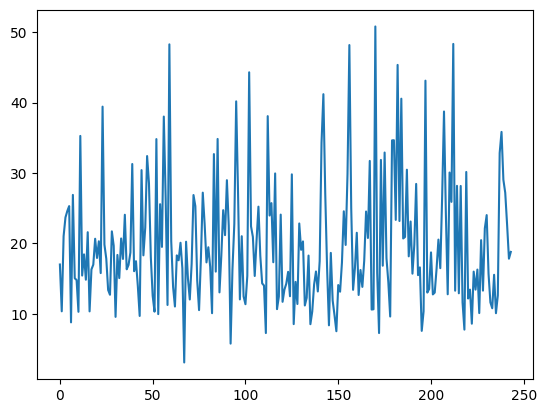
import matplotlib.pyplot as plt   
import pandas as pd   
import numpy as np   
  
import seaborn as sns

tip = sns.load\_dataset('tips')  
tip.head()

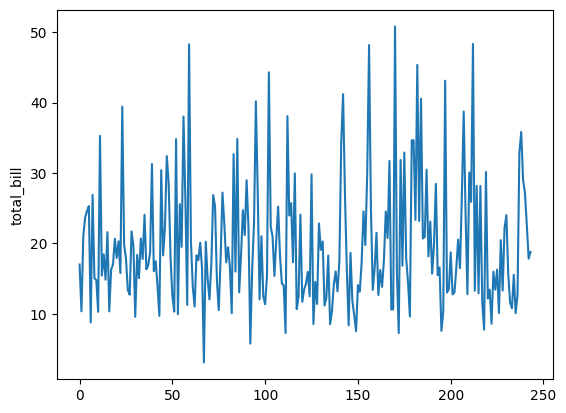
total\_bill tip sex smoker day time size  
0 16.99 1.01 Female No Sun Dinner 2  
1 10.34 1.66 Male No Sun Dinner 3  
2 21.01 3.50 Male No Sun Dinner 3  
3 23.68 3.31 Male No Sun Dinner 2  
4 24.59 3.61 Female No Sun Dinner 4

plt.plot(tip['total\_bill'])  
plt.show()



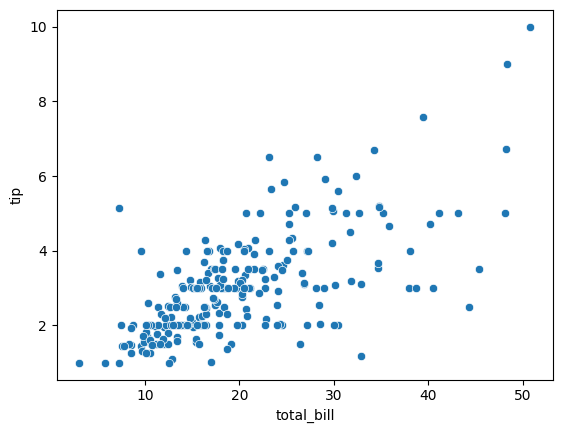
sns.lineplot(tip['total\_bill'])

<Axes: ylabel='total\_bill'>



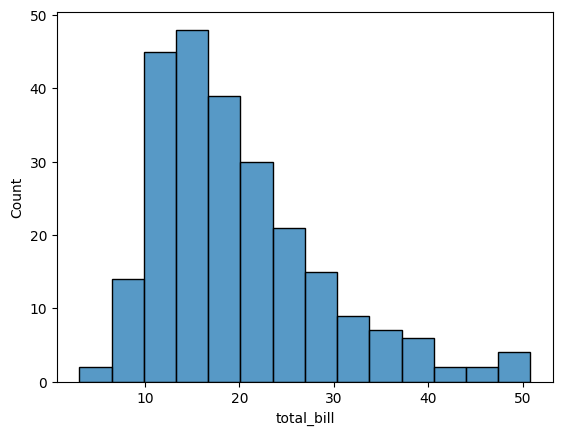
sns.scatterplot(x='total\_bill',y='tip',data=tip)

<Axes: xlabel='total\_bill', ylabel='tip'>



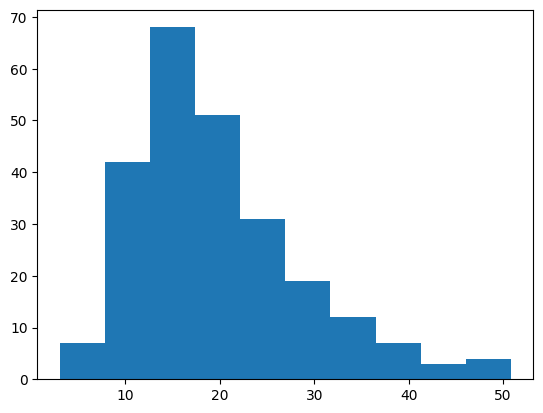
sns.histplot(tip['total\_bill'])

<Axes: xlabel='total\_bill', ylabel='Count'>



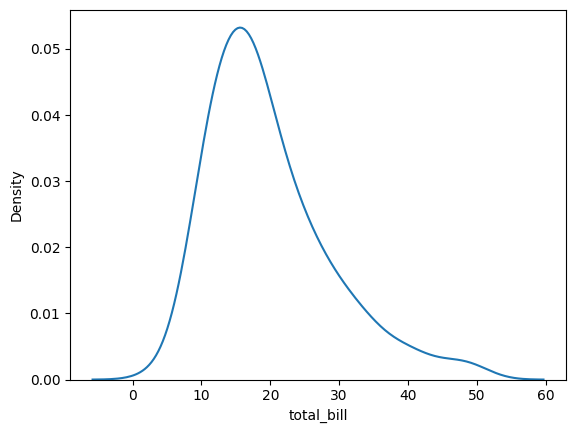
plt.hist(tip['total\_bill'])

(array([ 7., 42., 68., 51., 31., 19., 12., 7., 3., 4.]),  
 array([ 3.07 , 7.844, 12.618, 17.392, 22.166, 26.94 , 31.714, 36.488,  
 41.262, 46.036, 50.81 ]),  
 <BarContainer object of 10 artists>)



sns.kdeplot(tip['total\_bill'])

<Axes: xlabel='total\_bill', ylabel='Density'>



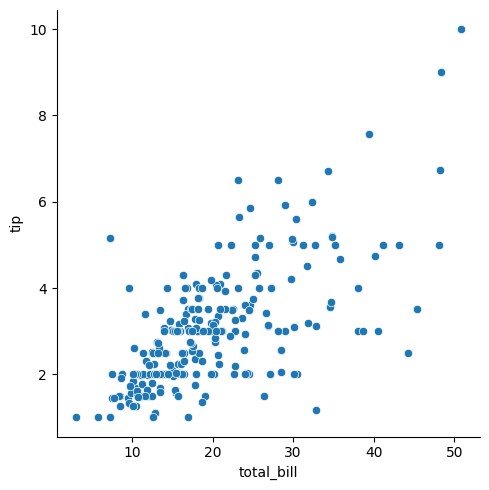
# catplot() --> categorical plot  
# relplot() --> relational plot

tip

total\_bill tip sex smoker day time size  
0 16.99 1.01 Female No Sun Dinner 2  
1 10.34 1.66 Male No Sun Dinner 3  
2 21.01 3.50 Male No Sun Dinner 3  
3 23.68 3.31 Male No Sun Dinner 2  
4 24.59 3.61 Female No Sun Dinner 4  
.. ... ... ... ... ... ... ...  
239 29.03 5.92 Male No Sat Dinner 3  
240 27.18 2.00 Female Yes Sat Dinner 2  
241 22.67 2.00 Male Yes Sat Dinner 2  
242 17.82 1.75 Male No Sat Dinner 2  
243 18.78 3.00 Female No Thur Dinner 2  
  
[244 rows x 7 columns]

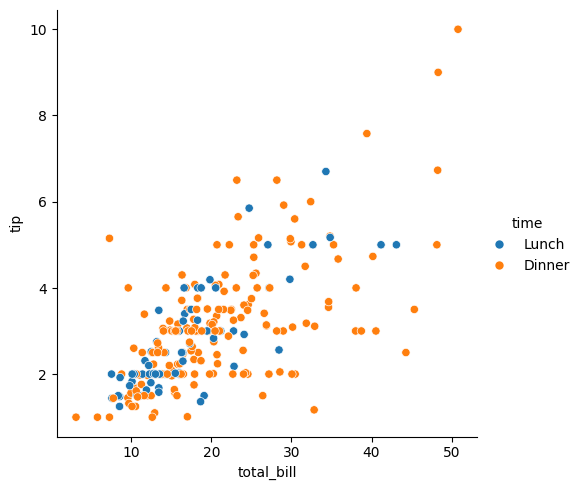
sns.relplot(x='total\_bill',y='tip',data=tip,kind='scatter')

<seaborn.axisgrid.FacetGrid at 0x25a37036f90>



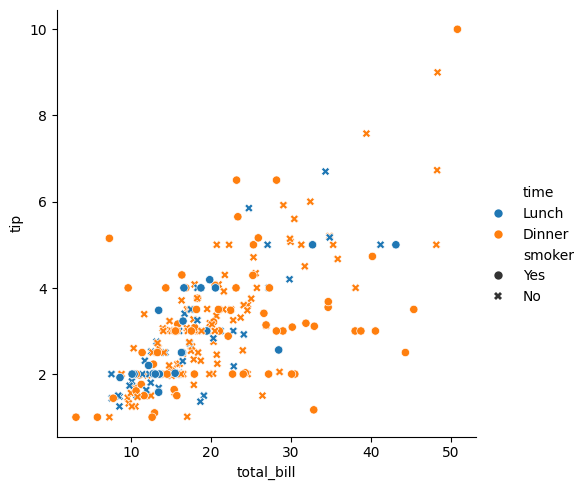
sns.relplot(x='total\_bill',y='tip',data=tip,kind='scatter',hue='time')

<seaborn.axisgrid.FacetGrid at 0x25a36f71070>



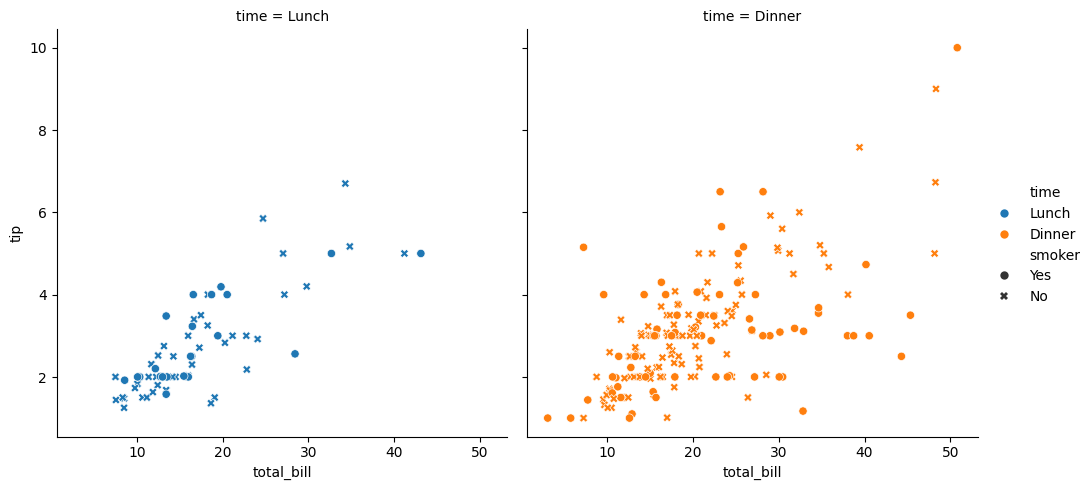
sns.relplot(x='total\_bill',y='tip',data=tip,kind='scatter',hue='time',style='smoker')

<seaborn.axisgrid.FacetGrid at 0x25a370fdf10>



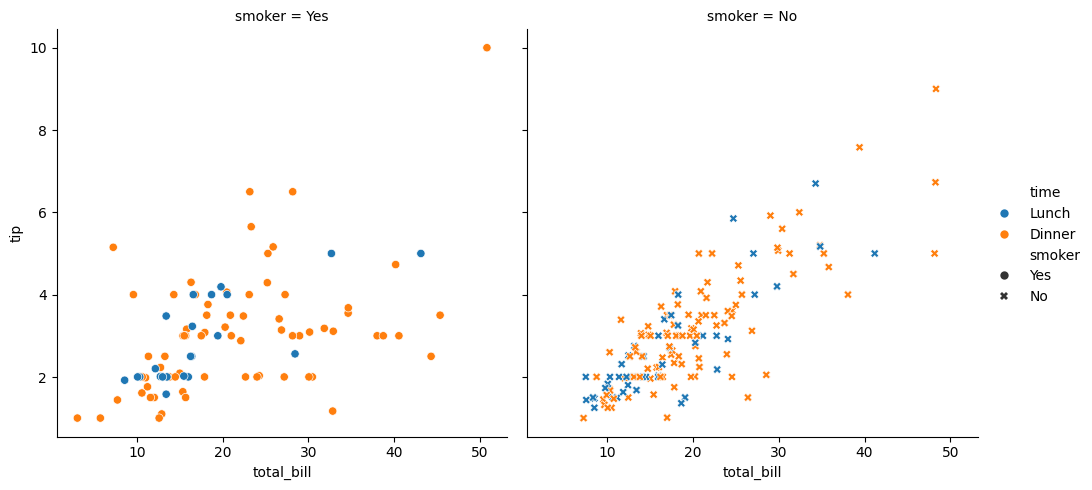
sns.relplot(x='total\_bill',y='tip',data=tip,kind='scatter',hue='time',style='smoker',col='time')

<seaborn.axisgrid.FacetGrid at 0x25a374079b0>



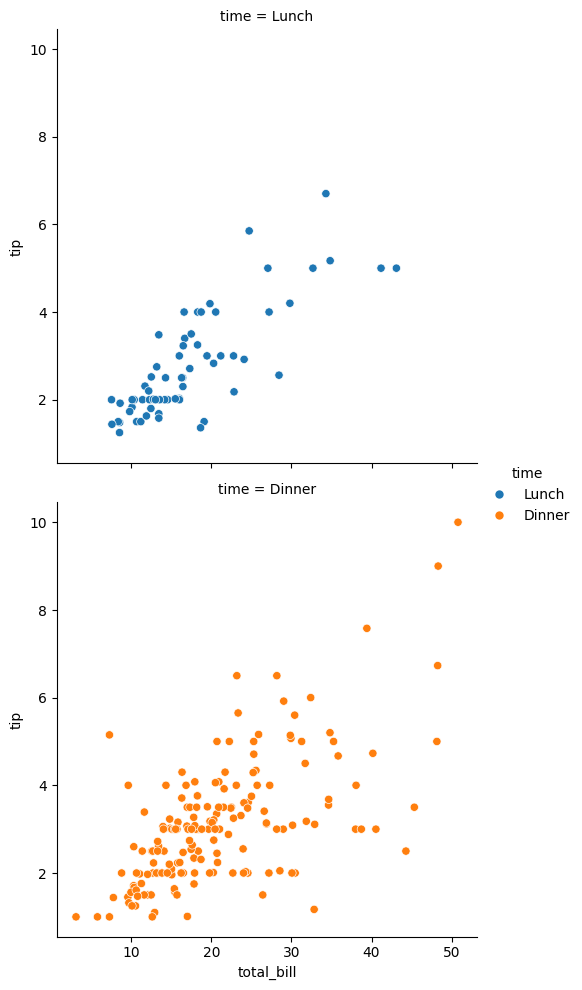
sns.relplot(x='total\_bill',y='tip',data=tip,kind='scatter',hue='time',style='smoker',col='smoker')

<seaborn.axisgrid.FacetGrid at 0x25a35ed28d0>



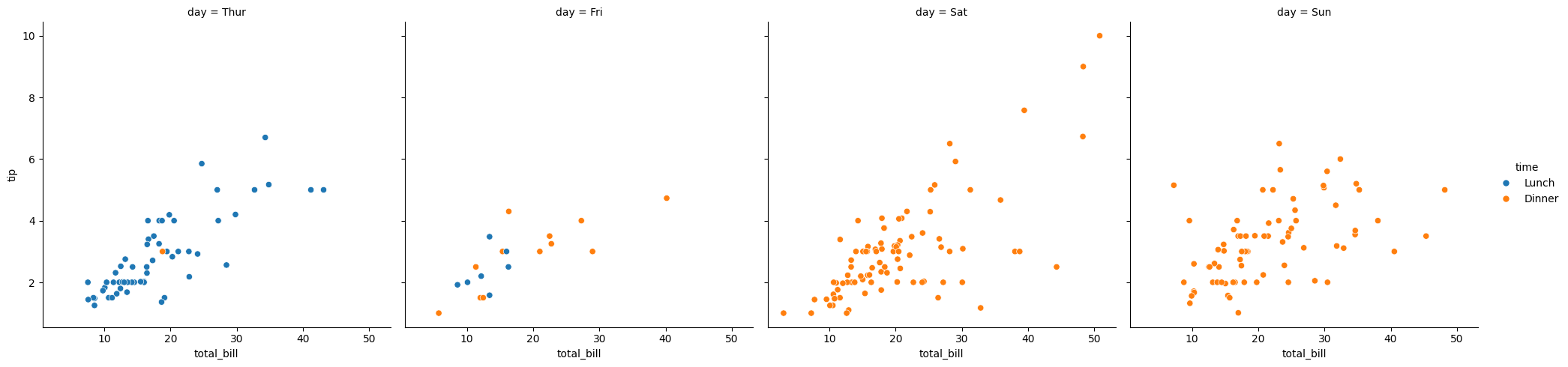
sns.relplot(x='total\_bill',y='tip',data=tip,kind='scatter',hue='time',row='time')

<seaborn.axisgrid.FacetGrid at 0x25a353dcb30>

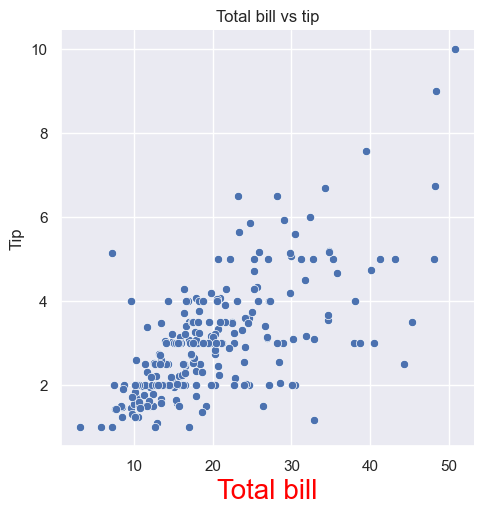


sns.relplot(x='total\_bill',y='tip',data=tip,kind='scatter',hue='time',col='day')

<seaborn.axisgrid.FacetGrid at 0x25a3abd28d0>

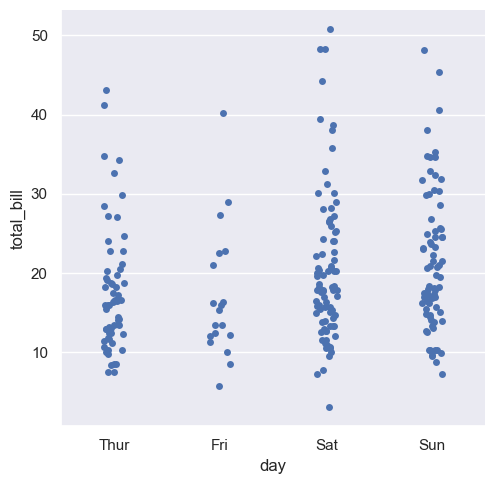


sns.relplot(x='total\_bill',y='tip',data=tip)  
plt.title('Total bill vs tip')  
plt.xlabel('Total bill',fontsize=20,color='red')  
plt.ylabel('Tip')  
plt.show()



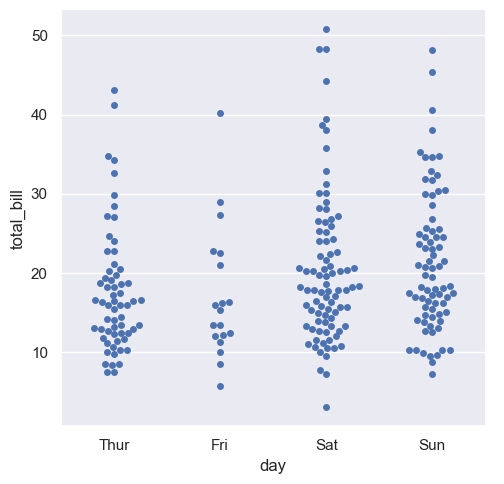
sns.catplot(x='day',y='total\_bill',data=tip,kind='strip')

<seaborn.axisgrid.FacetGrid at 0x25a3ccc3aa0>



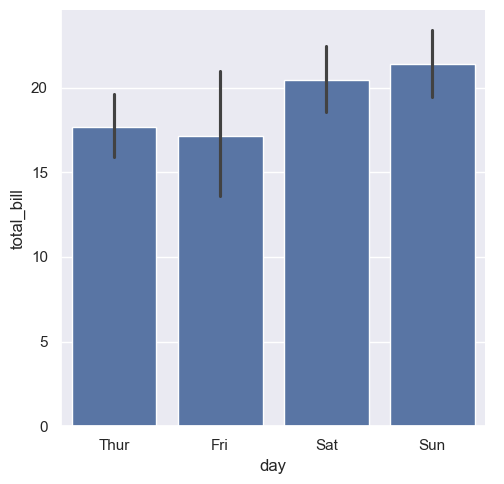
sns.catplot(x='day',y='total\_bill',data=tip,kind='swarm')

<seaborn.axisgrid.FacetGrid at 0x25a3acf2000>



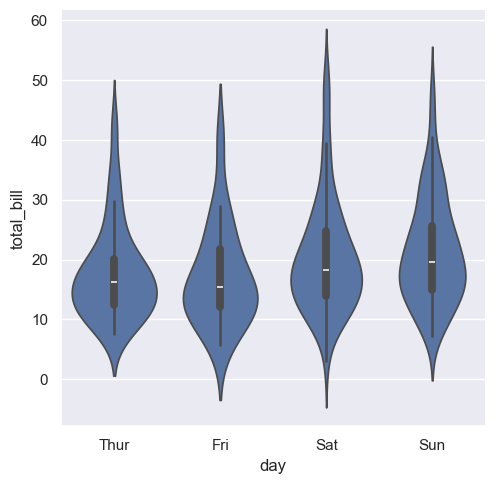
sns.catplot(x='day',y='total\_bill',data=tip,kind='bar')

<seaborn.axisgrid.FacetGrid at 0x25a3da761e0>



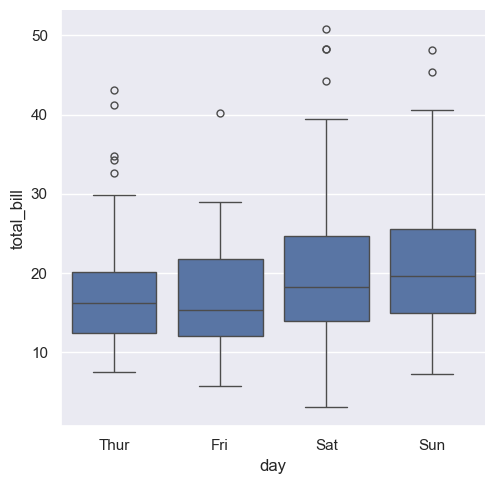
sns.catplot(x='day',y='total\_bill',data=tip,kind='violin')

<seaborn.axisgrid.FacetGrid at 0x25a401e2b70>



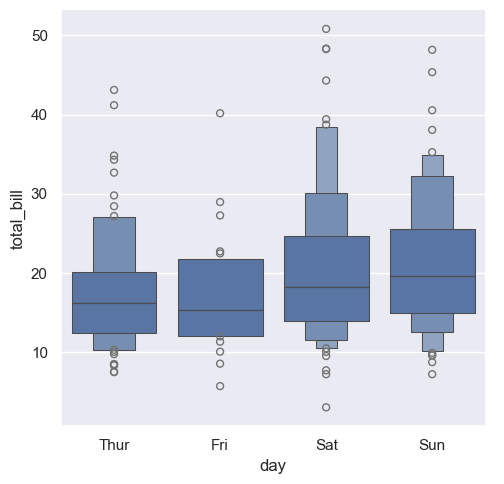
sns.catplot(x='day',y='total\_bill',data=tip,kind='box')  
sns.bar

<seaborn.axisgrid.FacetGrid at 0x25a3db98dd0>



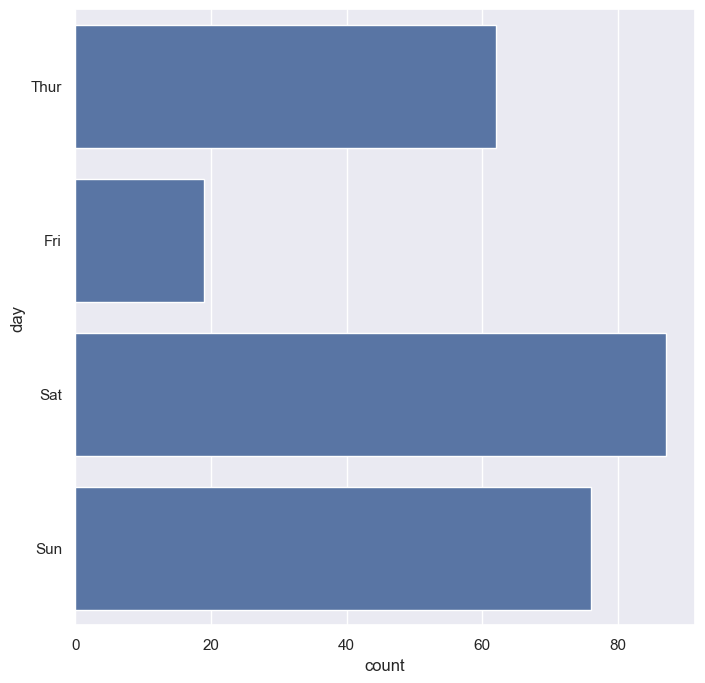
sns.catplot(x='day',y='total\_bill',data=tip,kind='boxen')

<seaborn.axisgrid.FacetGrid at 0x25a3dbd70b0>

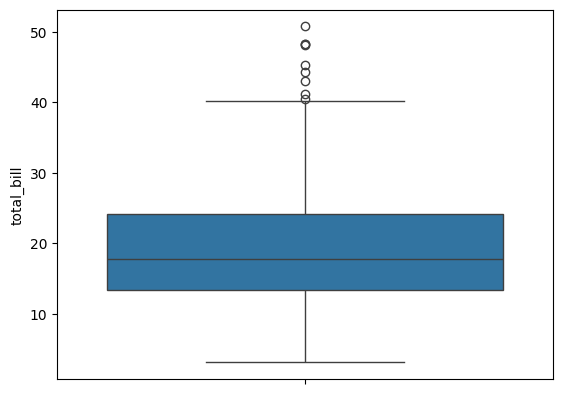


plt.figure(figsize=(8,8))  
sns.countplot(tip['day'])

<Axes: xlabel='count', ylabel='day'>



sns.boxplot(tip['total\_bill'])  
plt.show()



tip['total\_bill'].count()

244

tip.describe()  
  
# min , max , mean , count ,

total\_bill tip size  
count 244.000000 244.000000 244.000000  
mean 19.785943 2.998279 2.569672  
std 8.902412 1.383638 0.951100  
min 3.070000 1.000000 1.000000  
25% 13.347500 2.000000 2.000000  
50% 17.795000 2.900000 2.000000  
75% 24.127500 3.562500 3.000000  
max 50.810000 10.000000 6.000000

sorted\_total\_bill = tip.sort\_values(by='total\_bill')['total\_bill']  
sorted\_total\_bill

67 3.07  
92 5.75  
111 7.25  
172 7.25  
149 7.51  
 ...   
182 45.35  
156 48.17  
59 48.27  
212 48.33  
170 50.81  
Name: total\_bill, Length: 244, dtype: float64

sorted\_total\_bill[60:63]

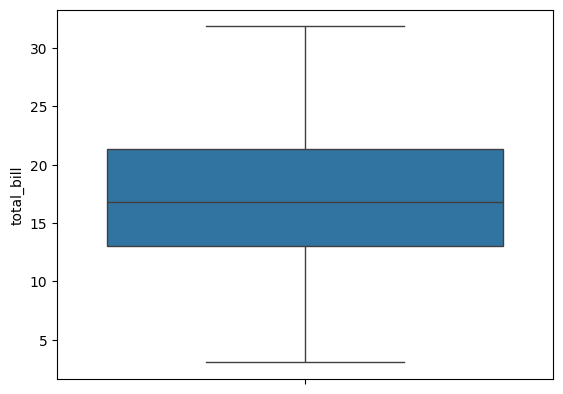
228 13.28  
26 13.37  
158 13.39  
Name: total\_bill, dtype: float64

### Dealing with outliers  
1. remove   
2. heart functional detection  
  
  
1 . remove  
2 . scale or transform

upper\_limit = tip['total\_bill'].quantile(0.90)

sns.boxplot(tip[tip['total\_bill']<upper\_limit]['total\_bill'])

<Axes: ylabel='total\_bill'>

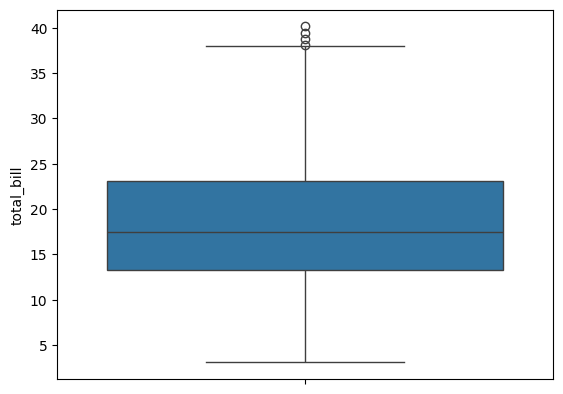


### using iqr range  
q1 = np.percentile(tip['total\_bill'],25)  
q3 = np.percentile(tip['total\_bill'],75)  
  
IQR = q3 - q1   
  
upper\_limit = q3 + 1.5 \* IQR  
lower\_limit = q1 - 1.5 \* IQR

outlier\_free\_bill = tip[(tip['total\_bill']<=upper\_limit) & (tip['total\_bill']>=lower\_limit)]

sns.boxplot(outlier\_free\_bill['total\_bill'])

<Axes: ylabel='total\_bill'>



import pandas as pd

tip.dtypes

total\_bill float64  
tip float64  
sex category  
smoker category  
day category  
time category  
size int64  
dtype: object

tip.select\_dtypes(include='category')

sex smoker day time  
0 Female No Sun Dinner  
1 Male No Sun Dinner  
2 Male No Sun Dinner  
3 Male No Sun Dinner  
4 Female No Sun Dinner  
.. ... ... ... ...  
239 Male No Sat Dinner  
240 Female Yes Sat Dinner  
241 Male Yes Sat Dinner  
242 Male No Sat Dinner  
243 Female No Thur Dinner  
  
[244 rows x 4 columns]

tip.select\_dtypes(exclude='category')

total\_bill tip size  
0 16.99 1.01 2  
1 10.34 1.66 3  
2 21.01 3.50 3  
3 23.68 3.31 2  
4 24.59 3.61 4  
.. ... ... ...  
239 29.03 5.92 3  
240 27.18 2.00 2  
241 22.67 2.00 2  
242 17.82 1.75 2  
243 18.78 3.00 2  
  
[244 rows x 3 columns]

IQR = q3-q1  
IQR

10.779999999999998