**Seaborn**

Types: 1.Figure Level 2.Axis Level

Classification: 1. Relational Plot 2. Distribution Plot 3. Categorical Plot 4. Regression Plot 5. Matrix plot 6. Multiplots

1. Relational Plot

Scatter Plot

```
import seaborn as sns
import matplotlib.pyplot as plt
import plotly.express as px
```

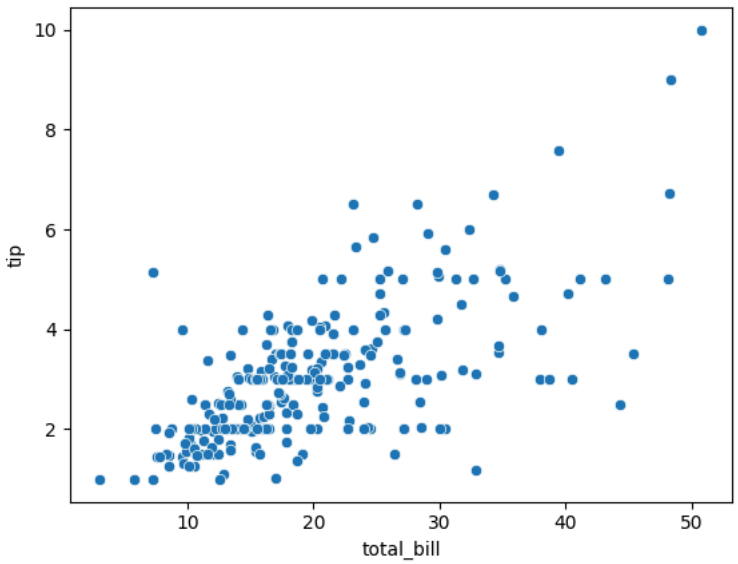
```
tips =sns.load_dataset('tips')
```

```
tips.head()
```

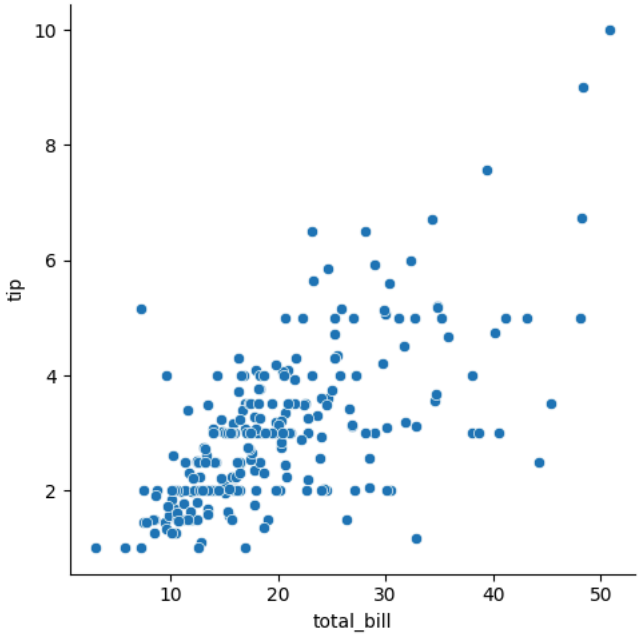
| | total_bill | tip | sex | smoker | day | time | size |
|---|---|---|---|---|---|---|---|
| **0** | 16.99 | 1.01 | Female | No | Sun | Dinner | 2 |
| **1** | 10.34 | 1.66 | Male | No | Sun | Dinner | 3 |
| **2** | 21.01 | 3.50 | Male | No | Sun | Dinner | 3 |
| **3** | 23.68 | 3.31 | Male | No | Sun | Dinner | 2 |
| **4** | 24.59 | 3.61 | Female | No | Sun | Dinner | 4 |

Next steps: Generate code with `tips` View recommended plots

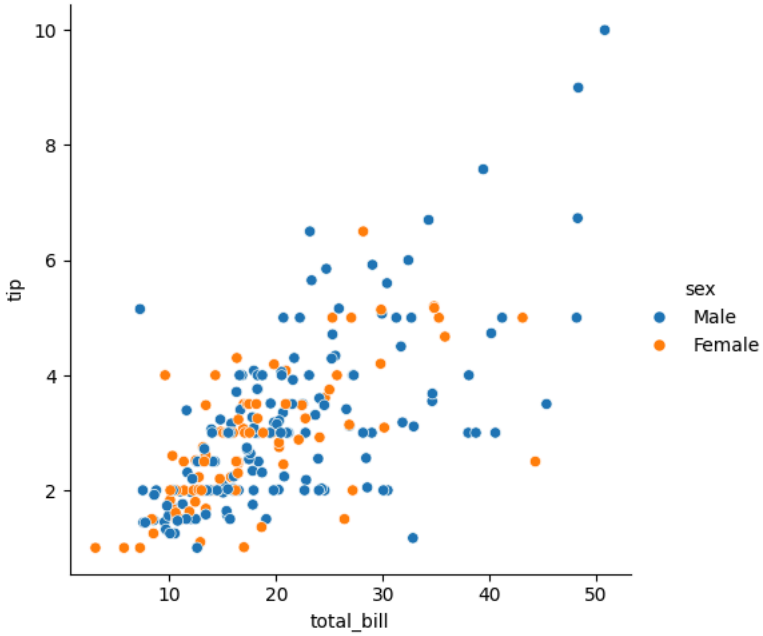
```
sns.scatterplot(data=tips,x='total_bill', y='tip')
plt.show()
```



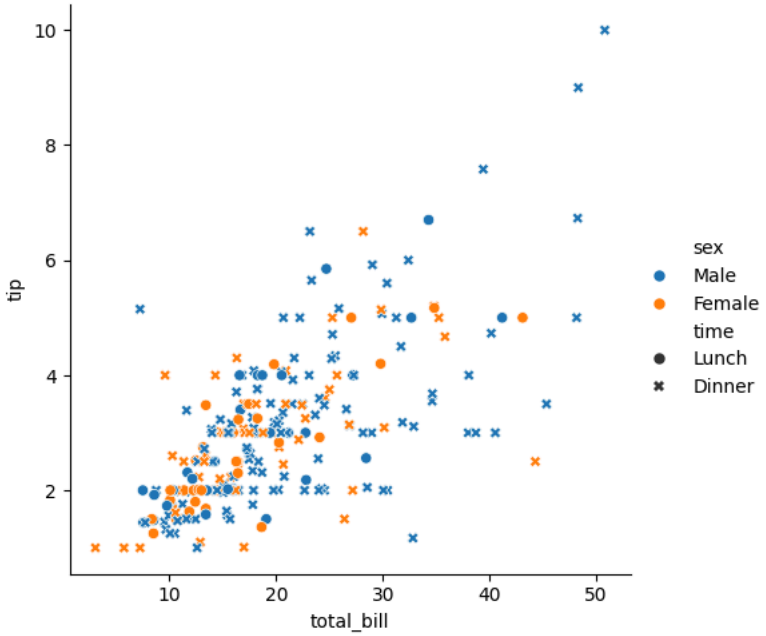

```
#relplot -> Figure Level ->Square Shape
sns.relplot(data=tips,x='total_bill', y='tip',kind="scatter")
plt.show()
```



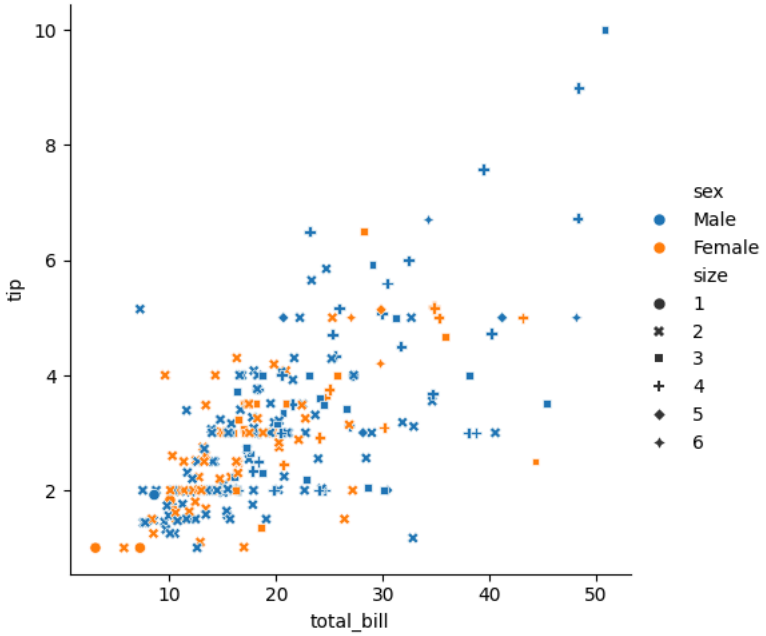

```
sns.relplot(data=tips,x='total_bill', y='tip',kind="scatter",hue='sex')
plt.show()
```



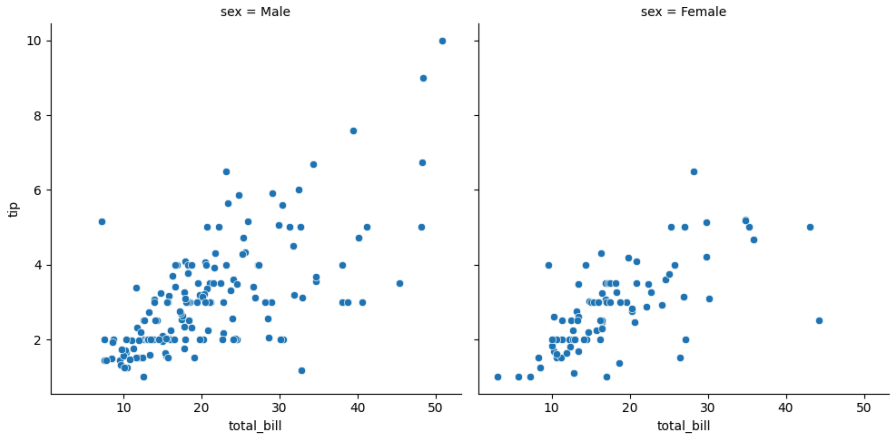

```
sns.relplot(data=tips,x='total_bill', y='tip',kind="scatter",hue='sex',style='time')
plt.show()
```



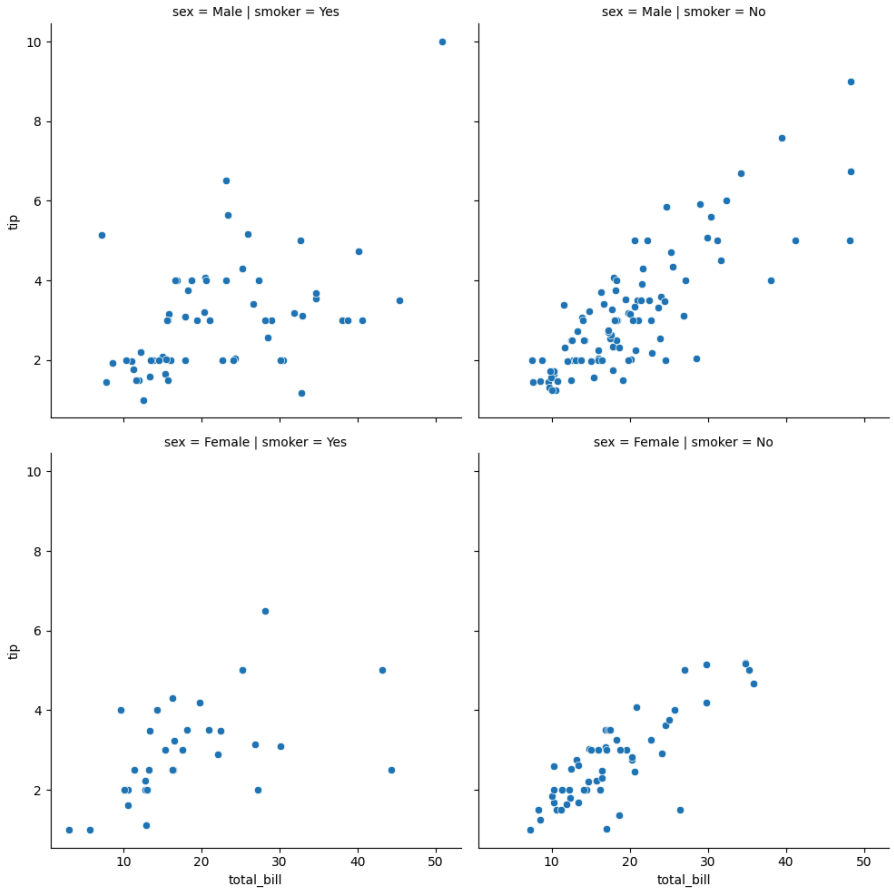

```
sns.relplot(data=tips,x='total_bill', y='tip',kind="scatter",hue='sex',style='size')
plt.show()
```



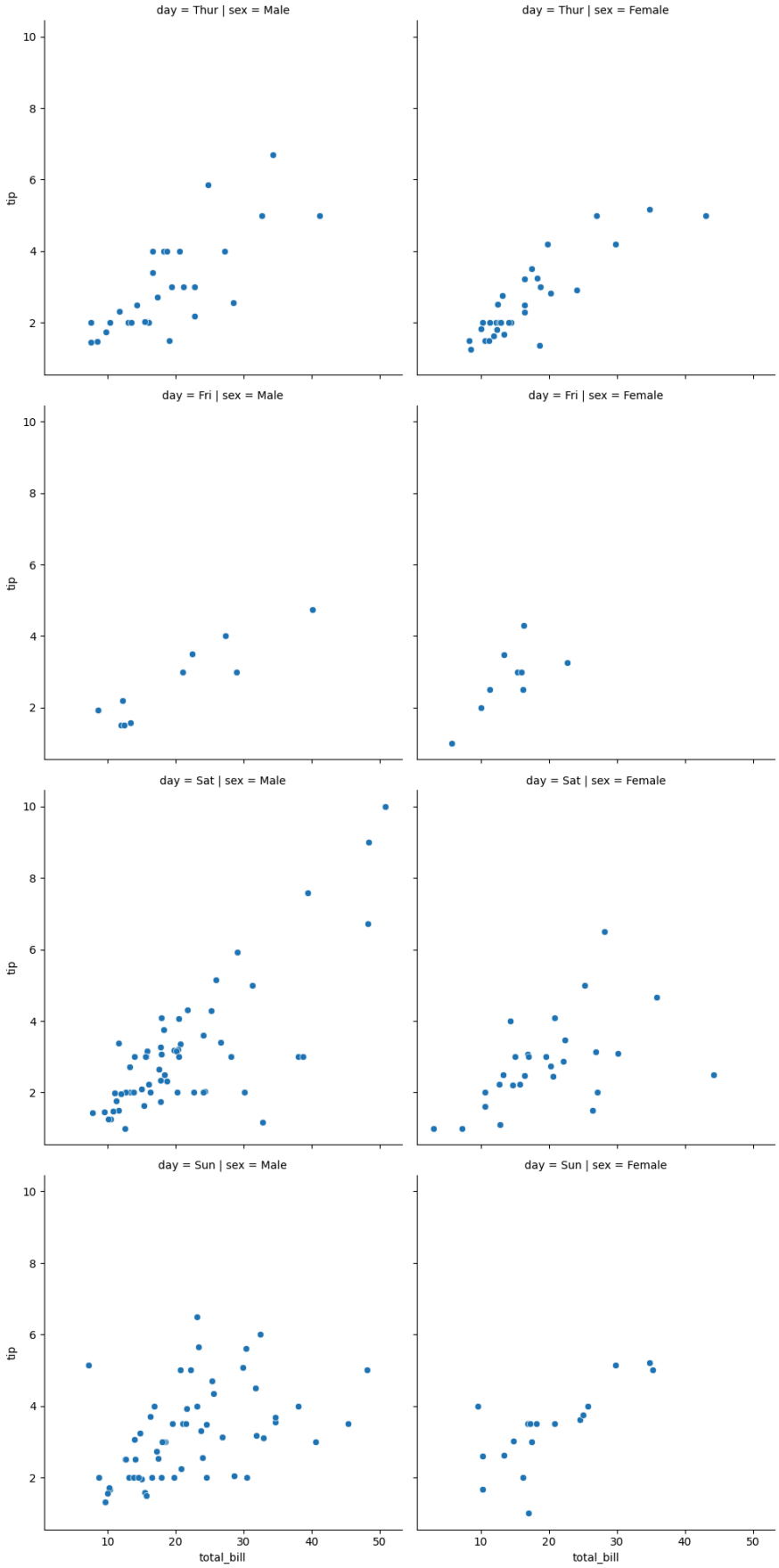

```
sns.relplot(data=tips,x='total_bill', y='tip',kind="scatter",col='sex')
plt.show()
```




```
sns.relplot(data=tips,x='total_bill', y='tip',kind="scatter",row='sex',col='smoker')
plt.show()
```



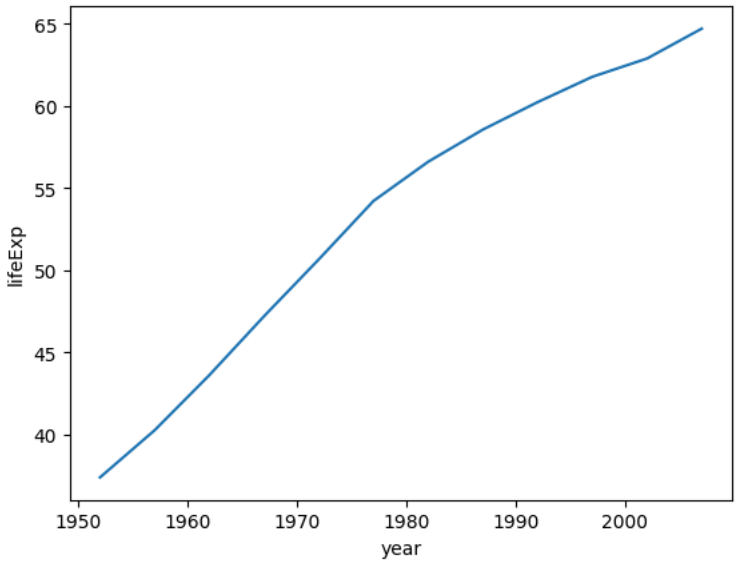

```
sns.relplot(data=tips,x='total_bill', y='tip',kind="scatter",row='day',col='sex')
plt.show()
```


day = Thur | sex = Male
day = Thur | sex = Female
day = Fri | sex = Male
day = Fri | sex = Female
day = Sat | sex = Male
day = Sat | sex = Female
day = Sun | sex = Male
day = Sun | sex = Female
tip
total_bill

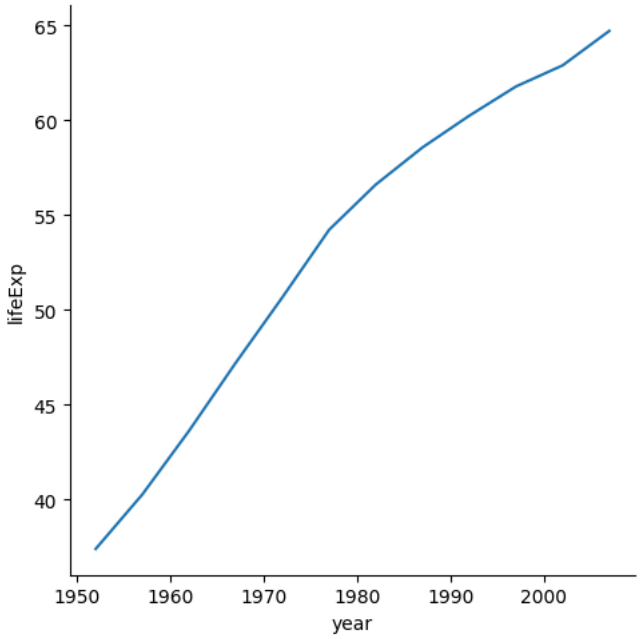
LinePlot

```
gap = px.data.gapminder()
gap.head()
temp =gap[gap['country']=="India"]
```

```
sns.lineplot(data=temp,x='year',y='lifeExp')
plt.show()
```




```
sns.relplot(data=temp,x='year',y='lifeExp',kind='line')
plt.show()
```

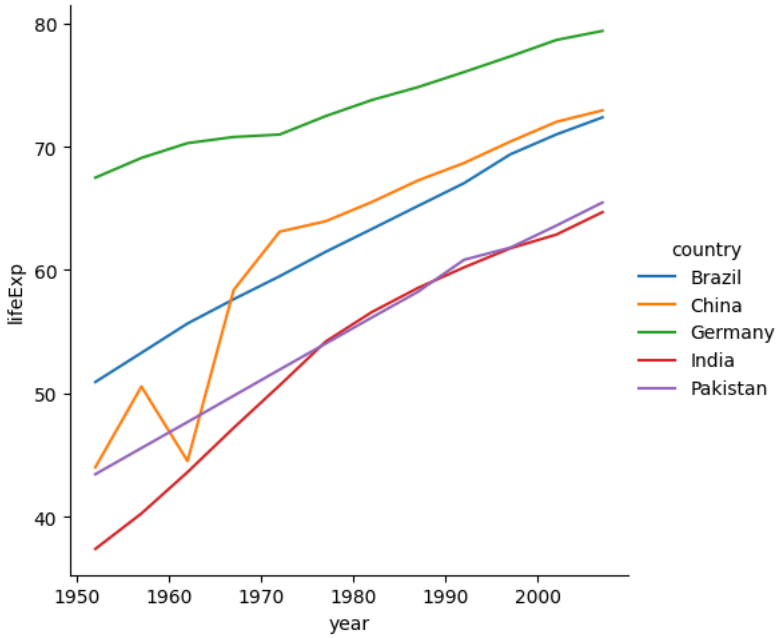



```
temp1=gap[gap['country'].isin(['India','Pakistan','China','Germany','Brazil'])]
temp1.head()
```

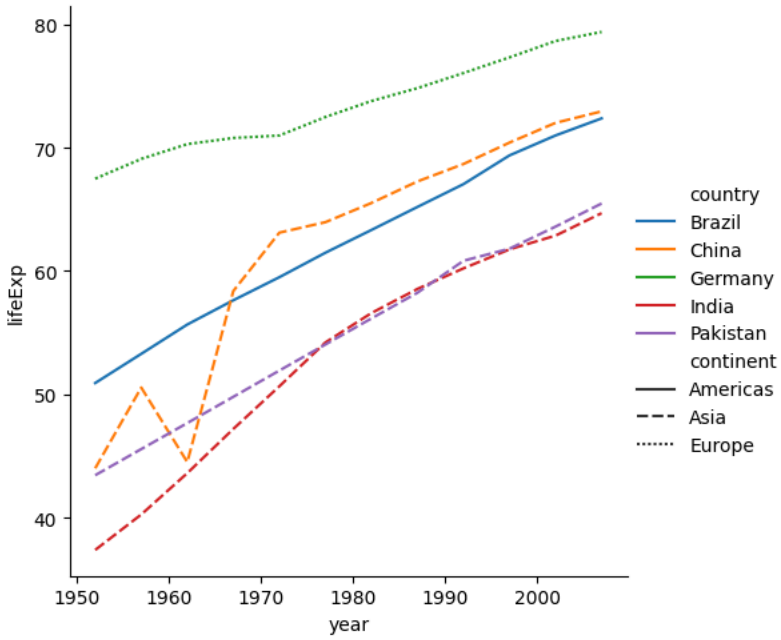
| | country | continent | year | lifeExp | pop | gdpPercap | iso_alpha | iso_num |
|---|---|---|---|---|---|---|---|---|
| **168** | Brazil | Americas | 1952 | 50.917 | 56602560 | 2108.944355 | BRA | 76 |
| **169** | Brazil | Americas | 1957 | 53.285 | 65551171 | 2487.365989 | BRA | 76 |
| **170** | Brazil | Americas | 1962 | 55.665 | 76039390 | 3336.585802 | BRA | 76 |
| **171** | Brazil | Americas | 1967 | 57.632 | 88049823 | 3429.864357 | BRA | 76 |
| **172** | Brazil | Americas | 1972 | 59.504 | 100840058 | 4985.711467 | BRA | 76 |

Next steps: Generate code with `temp1` View recommended plots

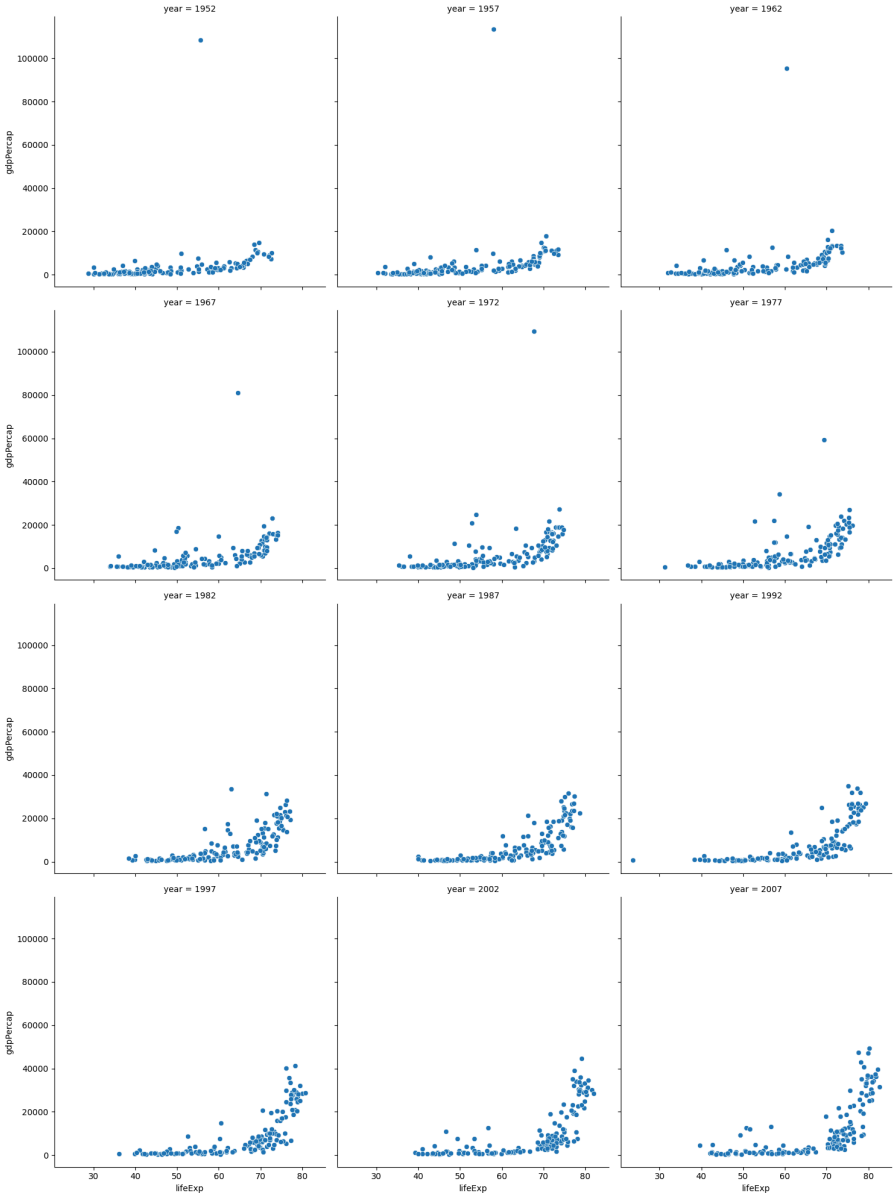
```
sns.relplot(data=temp1,x='year',y='lifeExp',kind='line',hue="country")
plt.show()
```



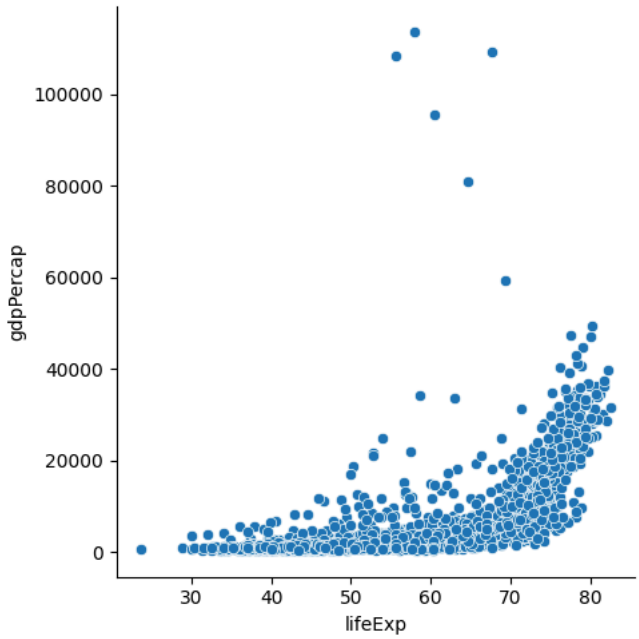

```
sns.relplot(data=temp1,x='year',y='lifeExp',kind='line',hue="country",style="continent")
plt.show()
```




```
sns.relplot(data=gap,x='lifeExp',y='gdpPercap',kind='scatter',col='year',col_wrap=3)
plt.show()
```


year = 1952
year = 1957
year = 1962
year = 1967
year = 1972
year = 1977
year = 1982
year = 1987
year = 1992
year = 1997
year = 2002
year = 2007
gdpPercap
lifeExp
0
20000
40000
60000
80000
100000
30
40
50
60
70
80

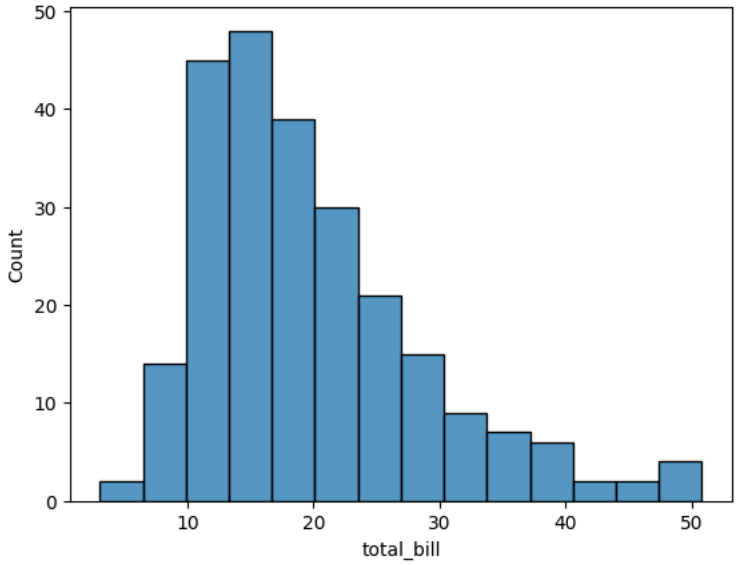
```
sns.relplot(data=gap,x='lifeExp',y='gdpPercap',kind='scatter')
plt.show()
```



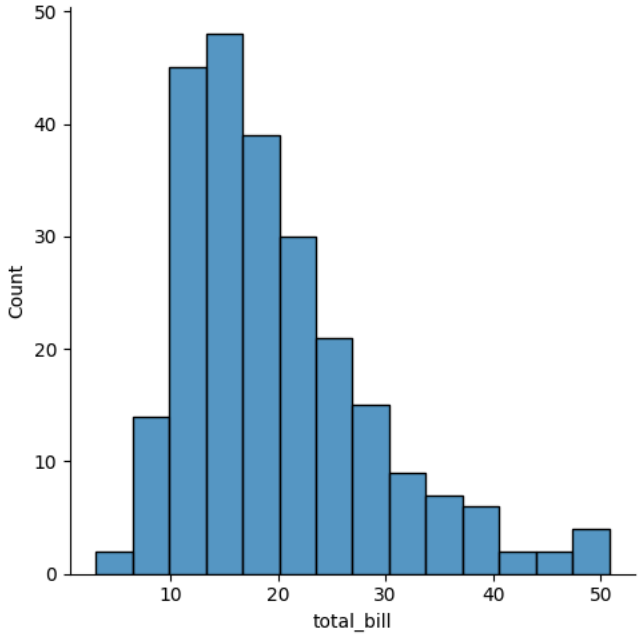

Distribution Plots

Histplot

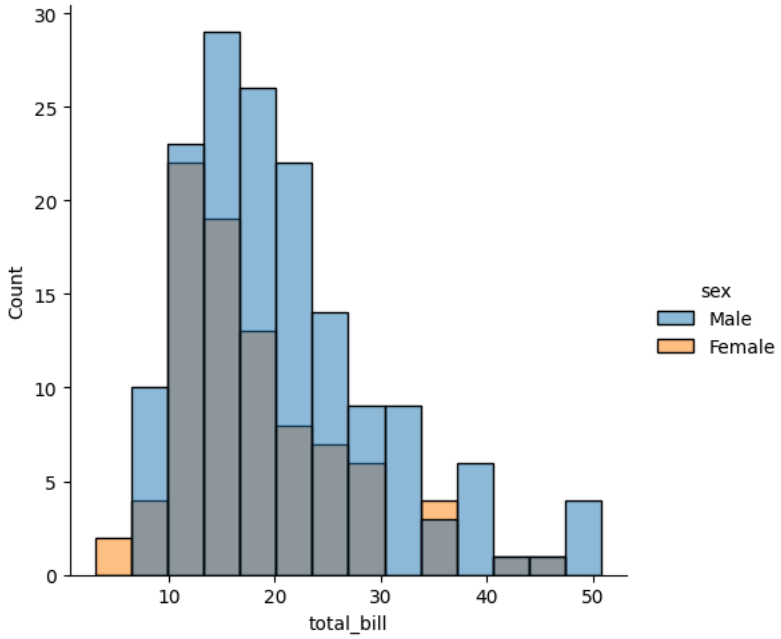
```
#Univariate histogram
sns.histplot(data=tips,x='total_bill')
plt.show()
```



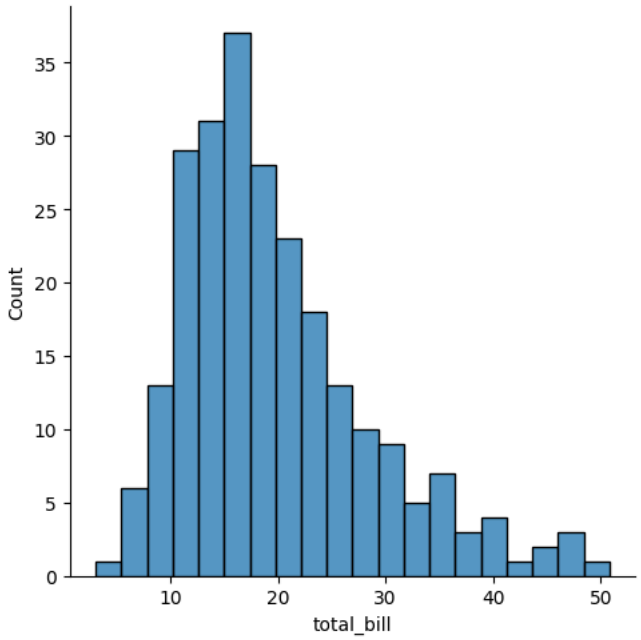

```
#Univariate histogram
sns.displot(kind='hist',data=tips,x='total_bill')
plt.show()
```



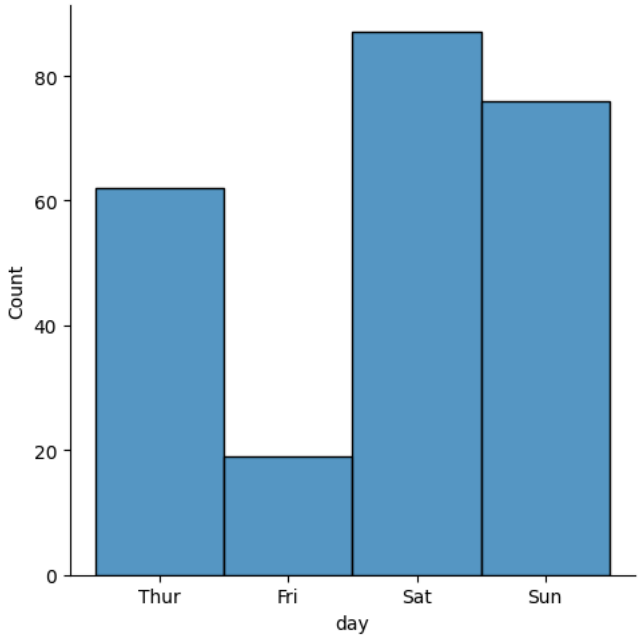

```
#hue
sns.displot(kind='hist',data=tips,x='total_bill',hue='sex')
plt.show()
```



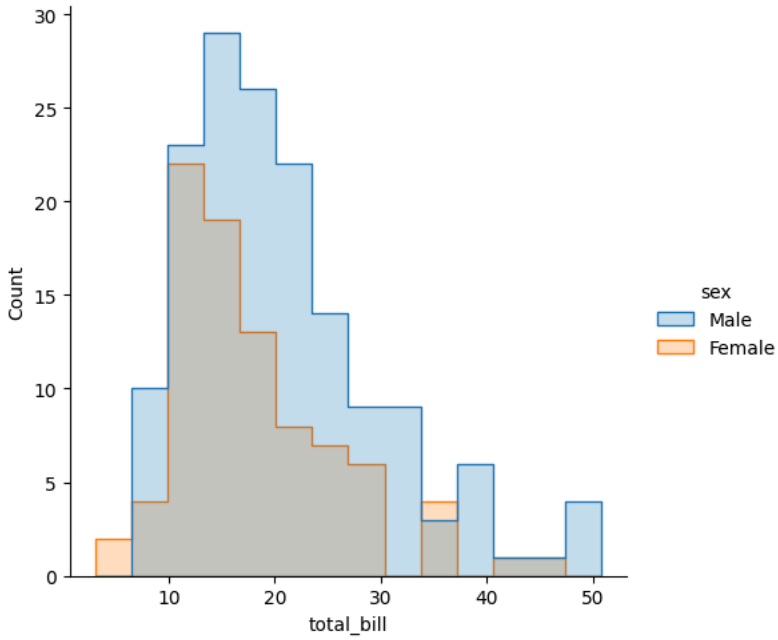

```
#bins
sns.displot(kind='hist',data=tips,x='total_bill',bins=20)
plt.show()
```




```
sns.displot(kind='hist',data=tips,x='day')
plt.show()
```




```
#element -> step
sns.displot(kind='hist',data=tips,x='total_bill',hue='sex',element='step')
plt.show()
```




```
titanic=sns.load_dataset('titanic')
titanic.head()
```

| | survived | pclass | sex | age | sibsp | parch | fare | embarked | class | who | adul |
|---|---|---|---|---|---|---|---|---|---|---|---|
| **0** | 0 | 3 | male | 22.0 | 1 | 0 | 7.2500 | S | Third | man | |
| **1** | 1 | 1 | female | 38.0 | 1 | 0 | 71.2833 | C | First | woman | |
| **2** | 1 | 3 | female | 26.0 | 0 | 0 | 7.9250 | S | Third | woman | |
| **3** | 1 | 1 | female | 35.0 | 1 | 0 | 53.1000 | S | First | woman | |
| **4** | 0 | 3 | male | 35.0 | 0 | 0 | 8.0500 | S | Third | man | |

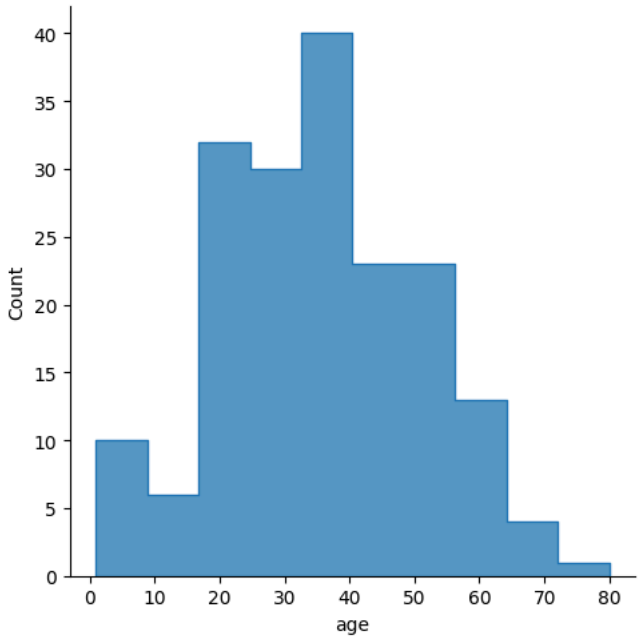
Next steps: Generate code with titanic View recommended plots

```
titanic.dropna(inplace=True)
```

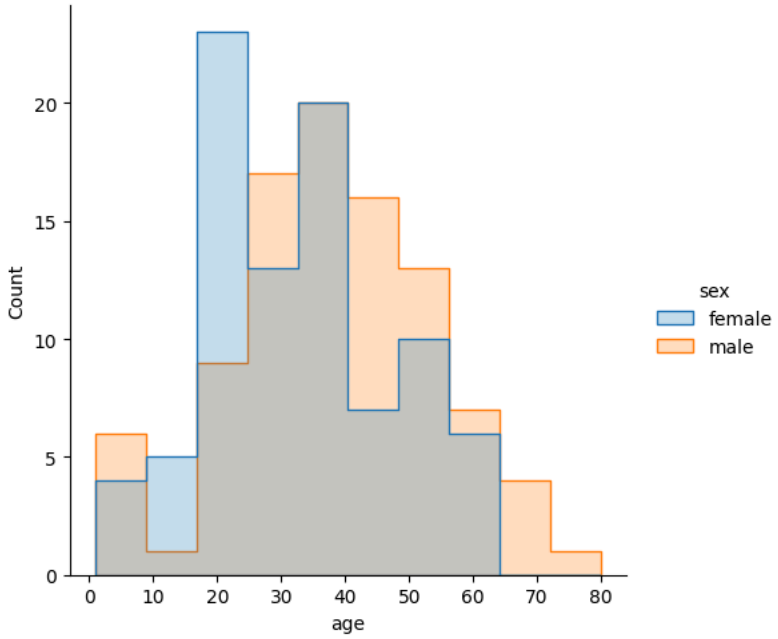
```
titanic.isnull().sum()
```

```
survived       0
pclass         0
sex            0
age            0
sibsp          0
parch          0
fare           0
embarked       0
class          0
who            0
adult_male     0
deck           0
embark_town    0
alive          0
alone          0
dtype: int64
```

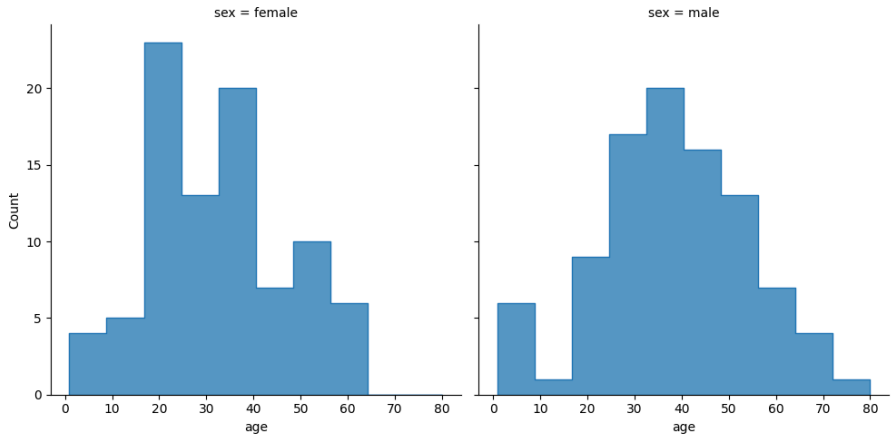
```
sns.displot(kind='hist',data=titanic,x='age',element='step')
plt.show()
```




```
sns.displot(kind='hist',data=titanic,x='age',hue='sex',element='step')
plt.show()
```

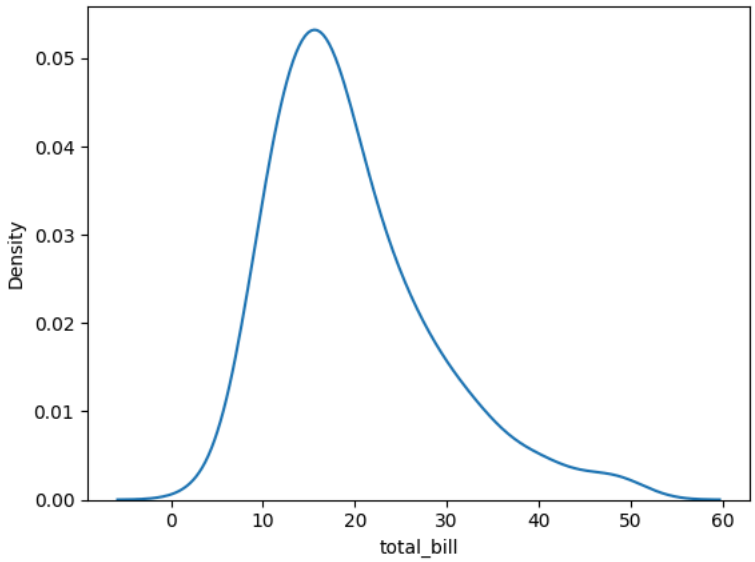



```
#faceting using row and col
sns.displot(kind='hist',data=titanic,x='age',element='step',col='sex')
plt.show()
```

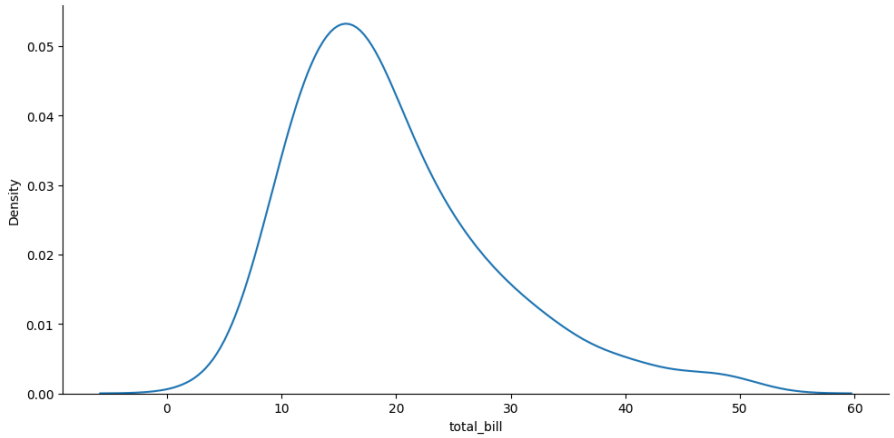



Kdeplot

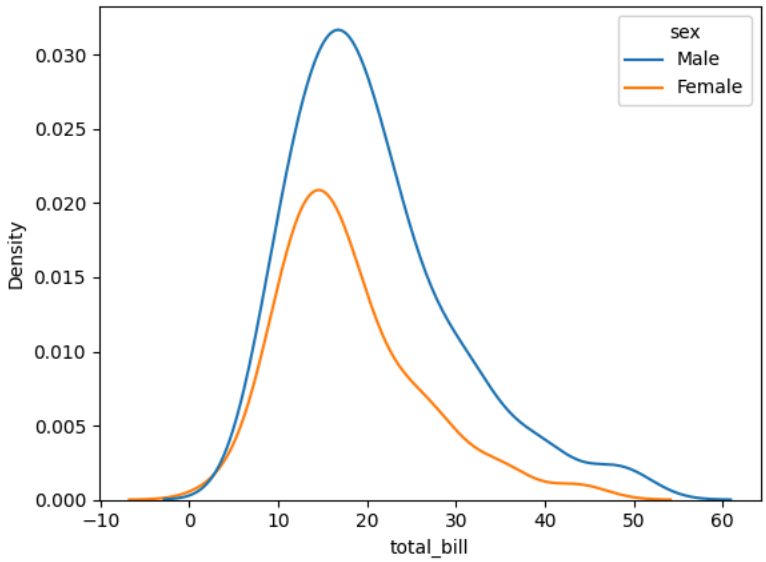
```
#kdeplot
sns.kdeplot(data=tips,x='total_bill')
plt.show()
```



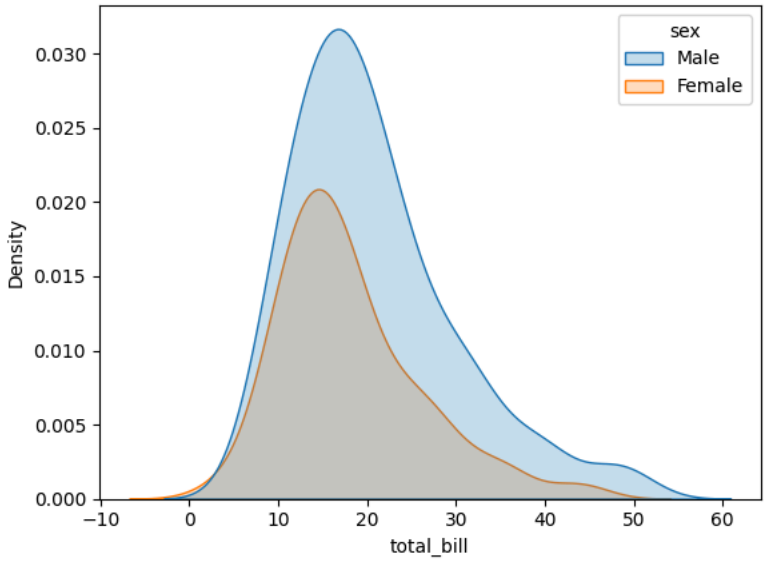

```
#kdeplot -> fig level
sns.displot(kind='kde',data=tips,x='total_bill',aspect=2)
plt.show()
```



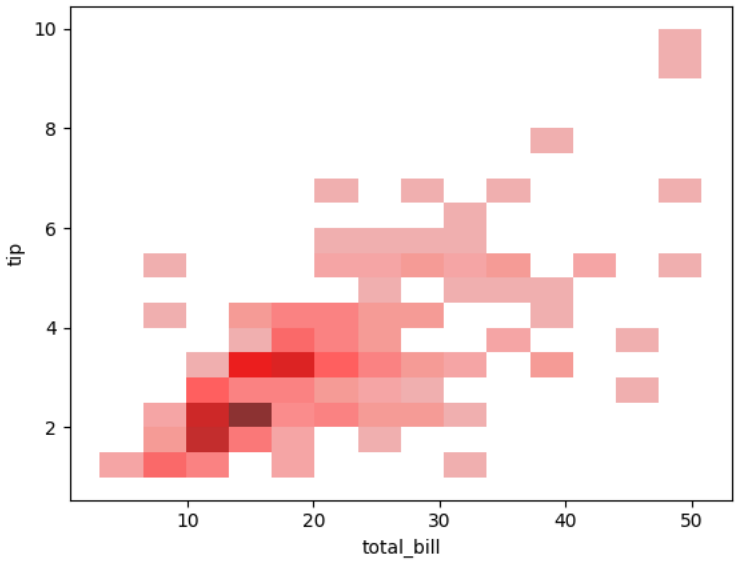

```
#hue
sns.kdeplot(data=tips,x='total_bill',hue='sex')
plt.show()
```



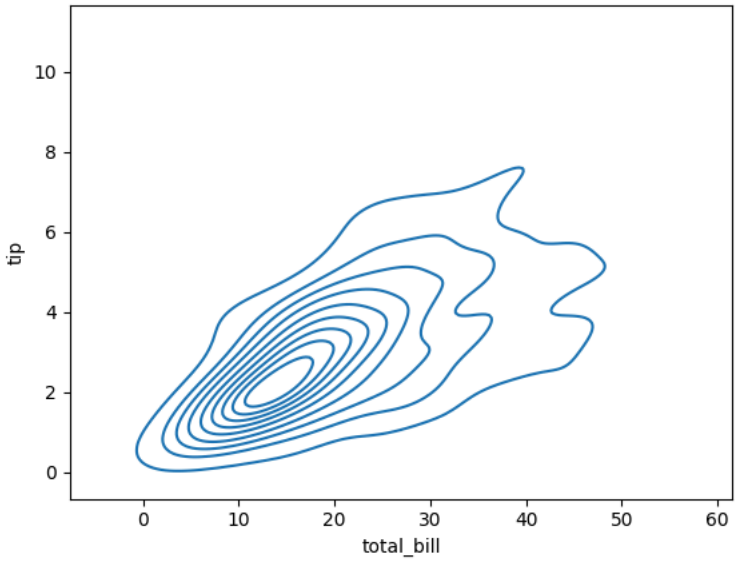

```
#hue -> fill
sns.kdeplot(data=tips,x='total_bill',hue='sex',fill=True)
plt.show()
```




```
#bivariate histogram
sns.histplot(data=tips,x='total_bill', y='tip',color='r')
plt.show()
```




```
#bivariate kdeplot
sns.kdeplot(data=tips,x='total_bill',y='tip')
plt.show()
```




Categorical Plots

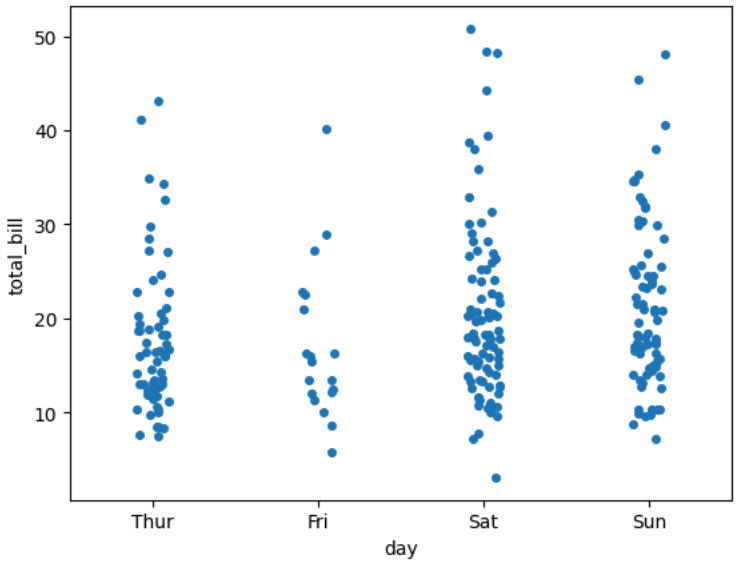
Categorical Scatter plot : 1.Stripplot 2.swarmplot

```
#Strip plot
tips.head()
```

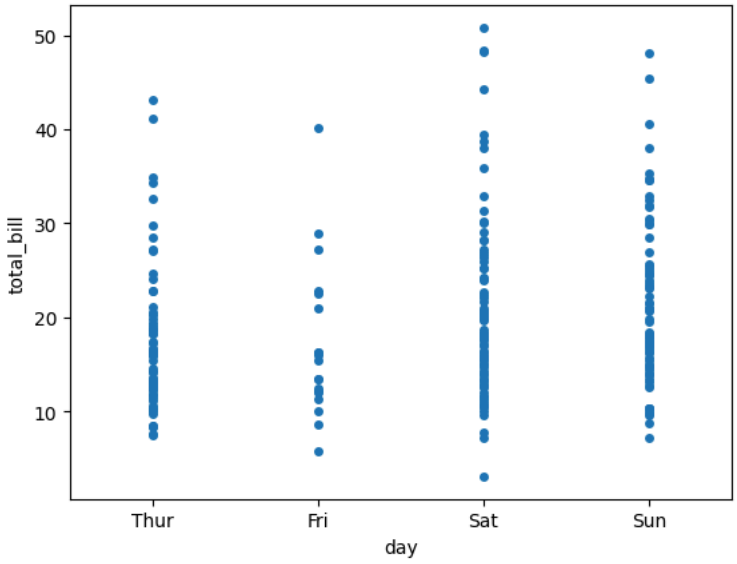
| | total_bill | tip | sex | smoker | day | time | size |
|---|---|---|---|---|---|---|---|
| **0** | 16.99 | 1.01 | Female | No | Sun | Dinner | 2 |
| **1** | 10.34 | 1.66 | Male | No | Sun | Dinner | 3 |
| **2** | 21.01 | 3.50 | Male | No | Sun | Dinner | 3 |
| **3** | 23.68 | 3.31 | Male | No | Sun | Dinner | 2 |
| **4** | 24.59 | 3.61 | Female | No | Sun | Dinner | 4 |

Next steps: Generate code with tips View recommended plots

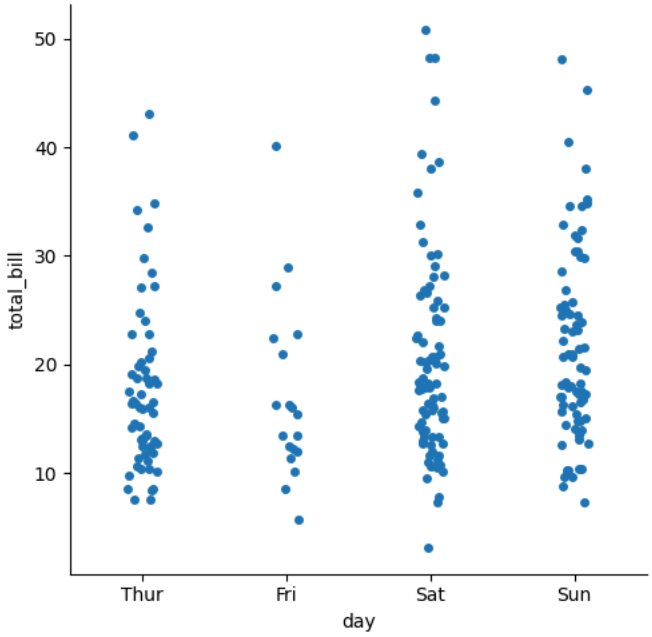
```
sns.stripplot(data=tips,x='day',y='total_bill')
plt.show()
```



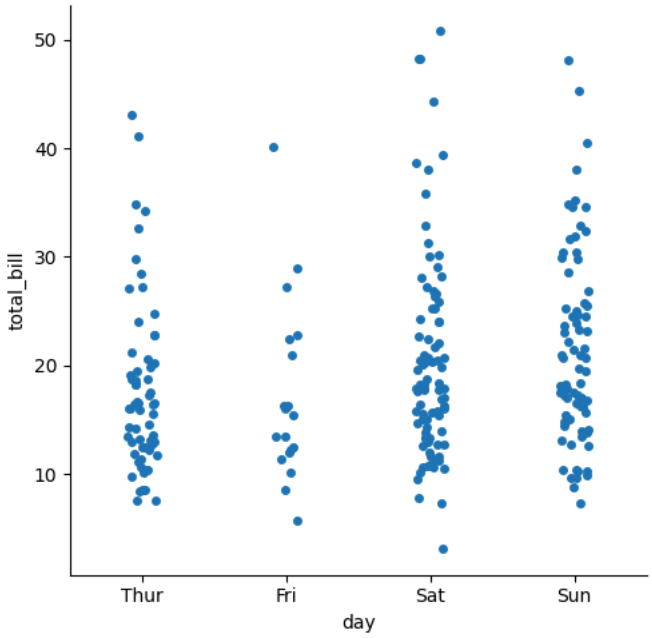

```
#jitter
sns.stripplot(data=tips,x='day',y='total_bill',jitter=False)
plt.show()
```



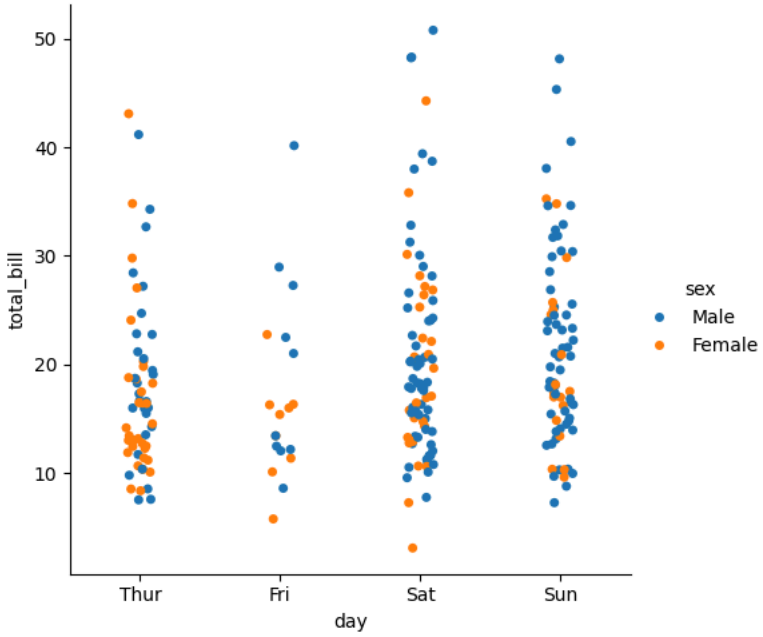

```
#using catplot ->fig. level

sns.catplot(kind="strip", data=tips, x='day', y='total_bill')
plt.show()
```



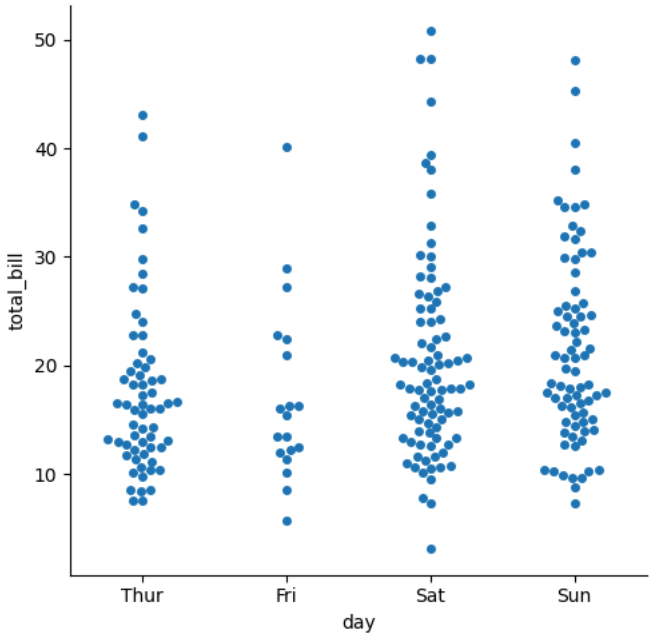

```
#jitter
sns.catplot(kind="strip", data=tips, x='day', y='total_bill',jitter=0.1)
plt.show()
```



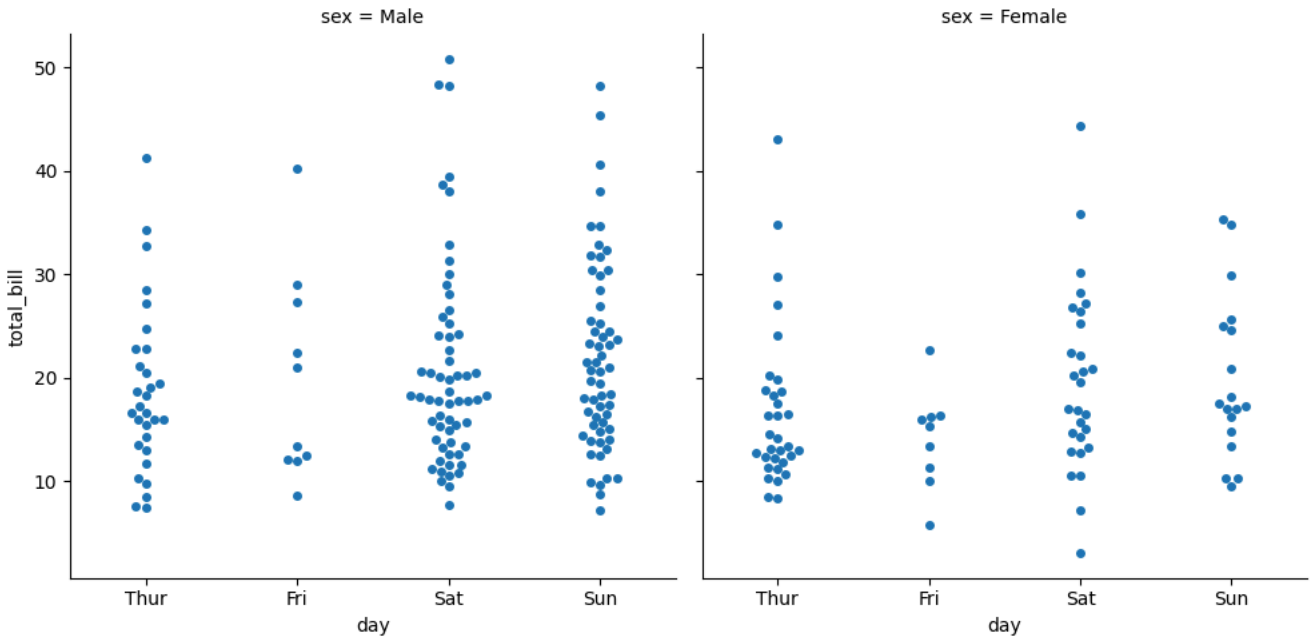

```
#hue
sns.catplot(kind="strip", data=tips, x='day', y='total_bill',hue='sex')
plt.show()
```




```
#swarmplot
sns.catplot(kind="swarm", data=tips, x='day', y='total_bill')
plt.show()
```

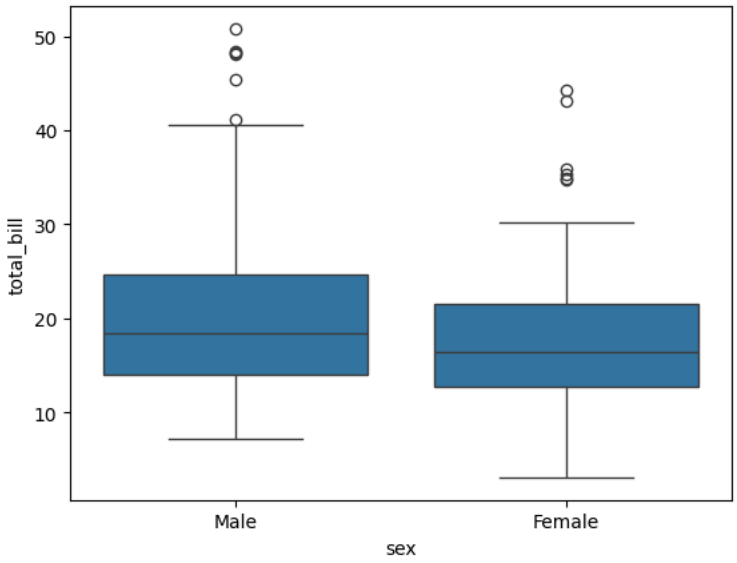



```
#hue
sns.catplot(kind="swarm", data=tips, x='day', y='total_bill',col='sex')
plt.show()
```

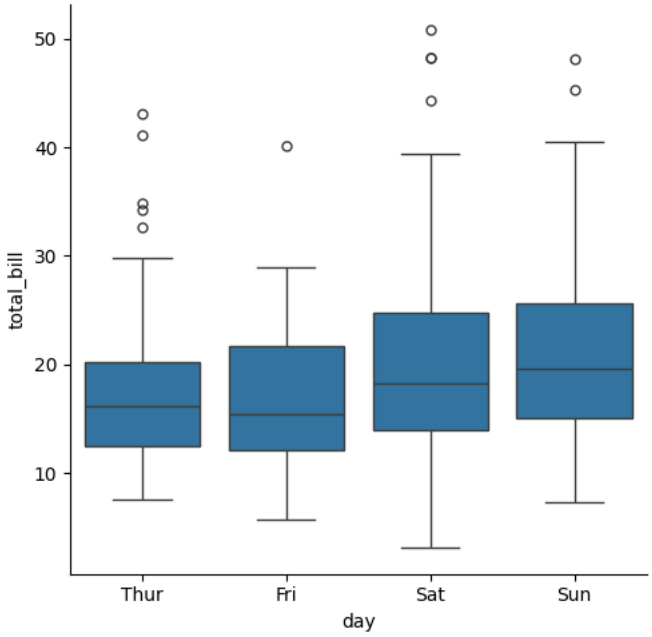



2.Distribution : 1.Boxplot 2.Violinplot

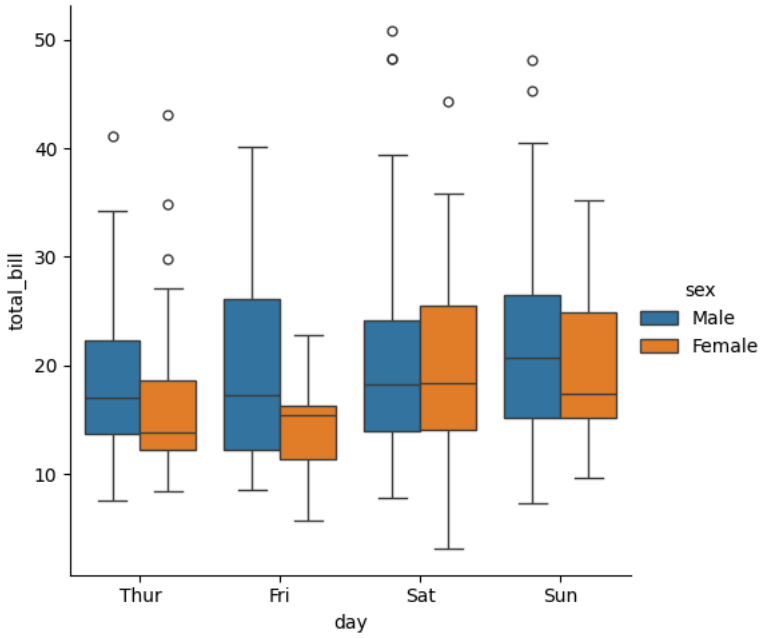
```
#Boxplot
sns.boxplot(data=tips,x='sex',y='total_bill')
plt.show()
```



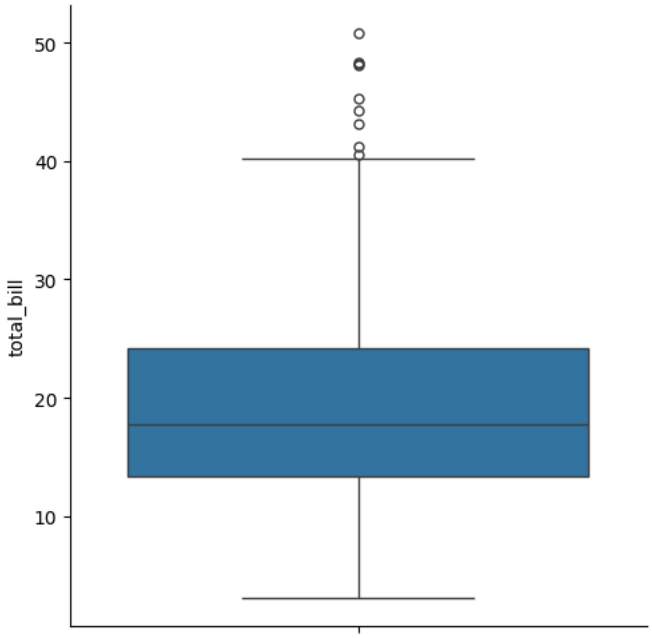

```
sns.catplot(kind='box',data=tips,x='day',y='total_bill')
plt.show()
```



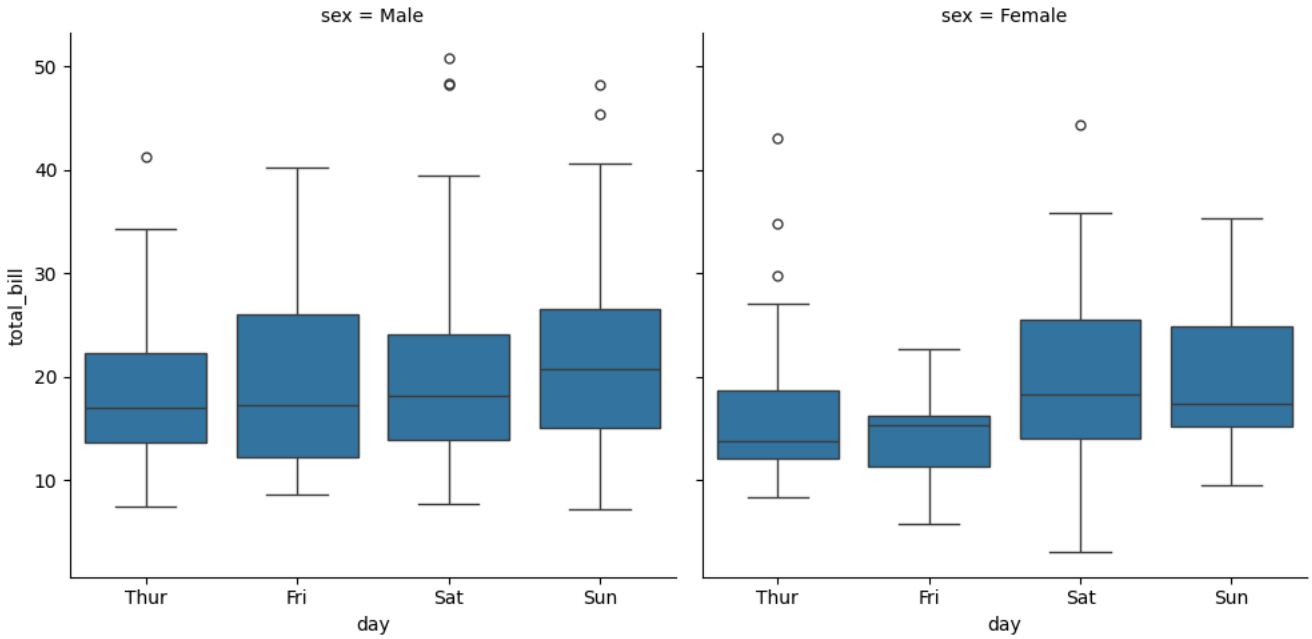

```
#hue
sns.catplot(kind='box',data=tips,x='day',y='total_bill',hue='sex')
plt.show()
```



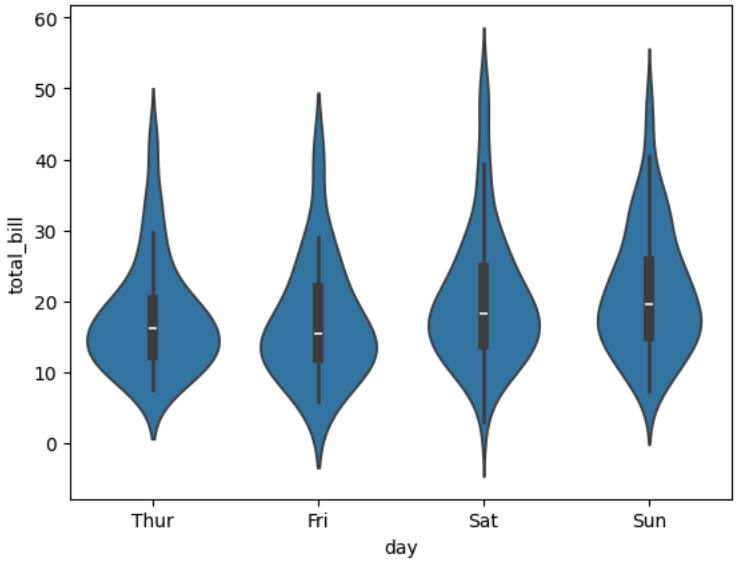

```
sns.catplot(kind='box',data=tips,y='total_bill')
plt.show()
```



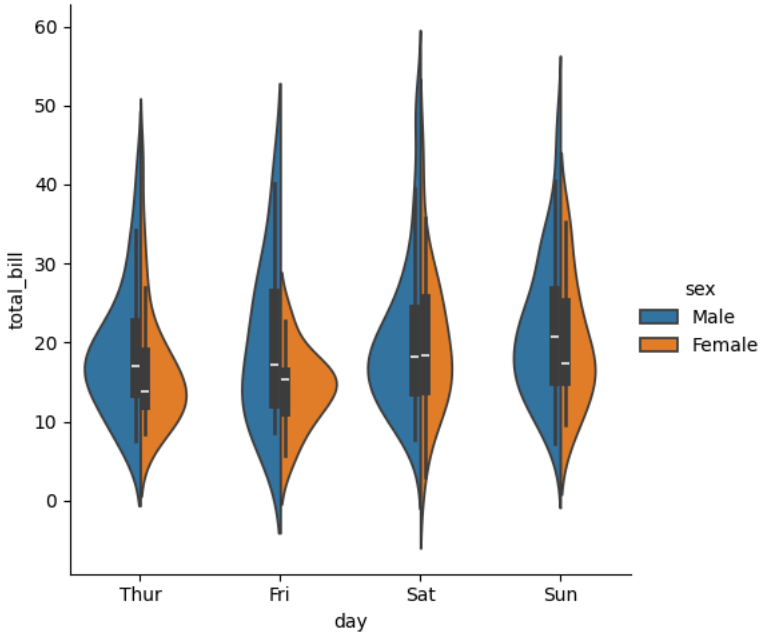

```
#col
sns.catplot(kind='box',data=tips,x='day',y='total_bill',col='sex')
plt.show()
```




```
#Violinplot =(Boxplot + KDEplot)
sns.violinplot(data=tips,x='day',y='total_bill')
plt.show()
```

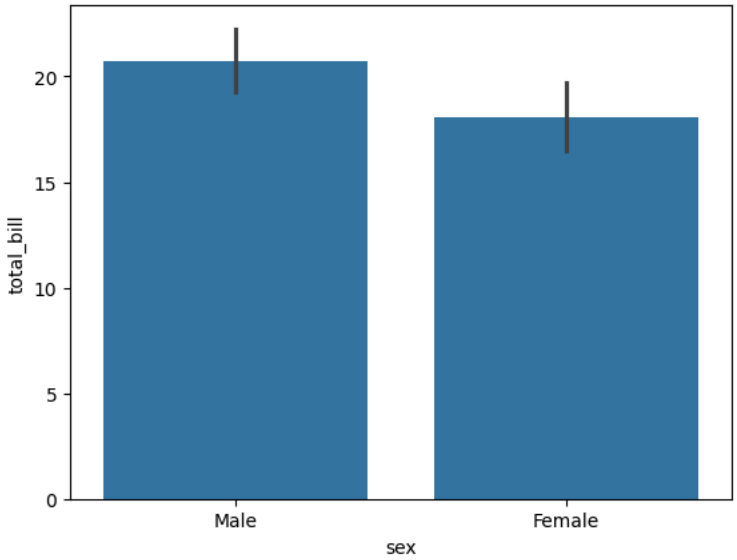



```
#hue
sns.catplot(kind='violin',data=tips,x='day',y='total_bill',hue='sex',split=True)
plt.show()
```

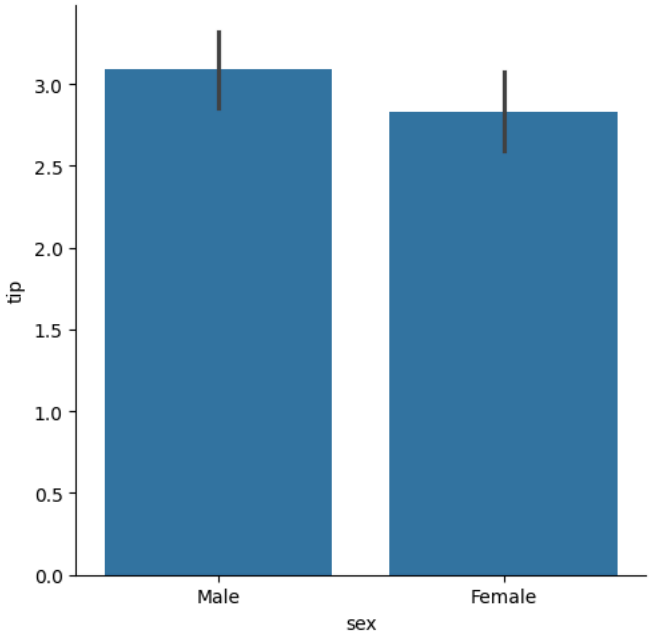



Estimate plot : 1.Barplot 2.Pointplot 3.Countplot

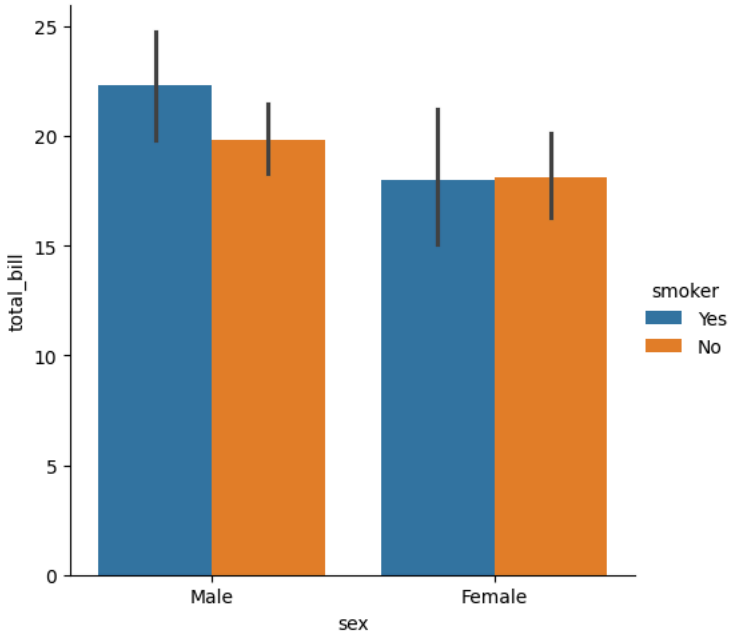
```
#barplot
sns.barplot(data=tips,x='sex',y='total_bill')
plt.show()
```



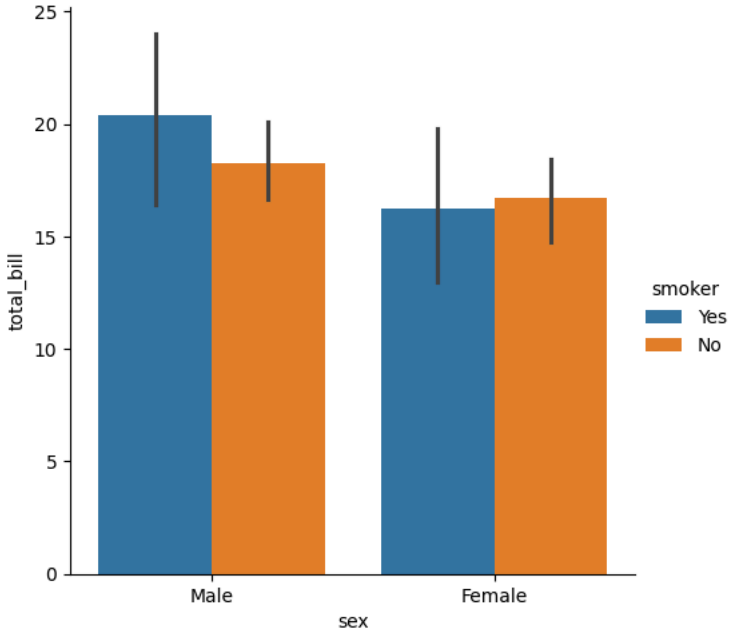

```
#barplot
sns.catplot(data=tips,x='sex',y='tip',kind='bar')
plt.show()
```



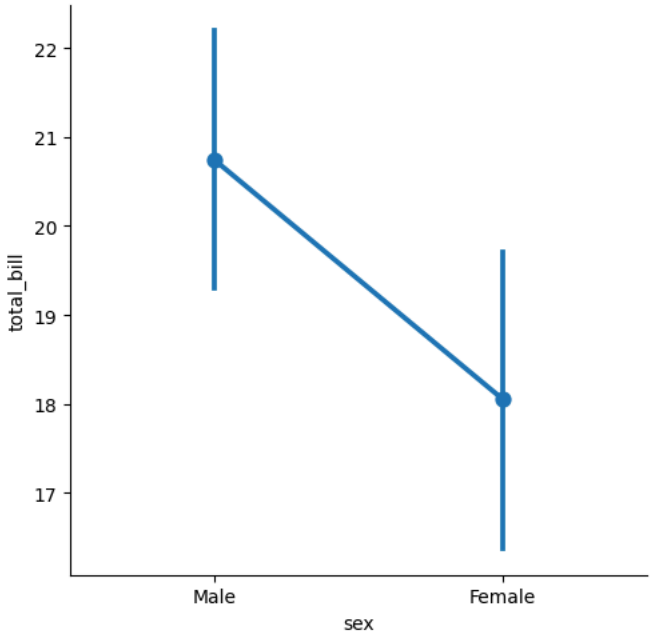

```
#hue
sns.catplot(kind='bar',data=tips,x='sex',y='total_bill',hue='smoker')
plt.show()
```



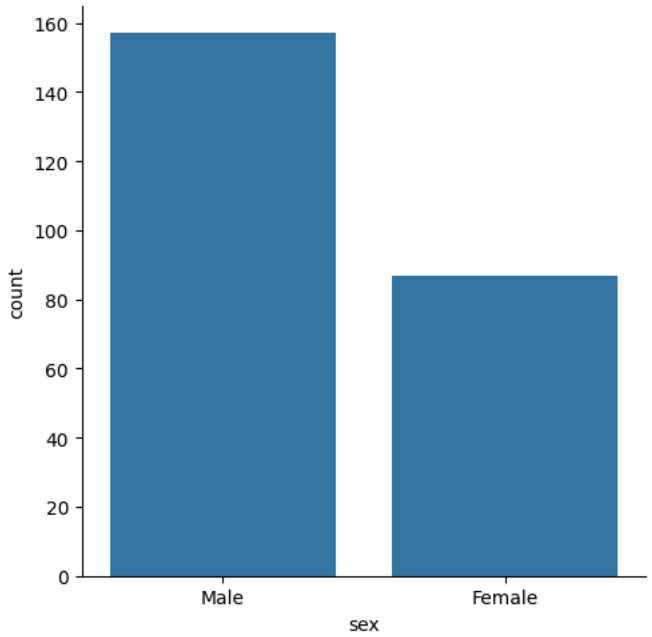

```
#estimator
import numpy as np
sns.catplot(kind='bar',data=tips,x='sex',y='total_bill',hue='smoker',estimator=np.median)
plt.show()
```



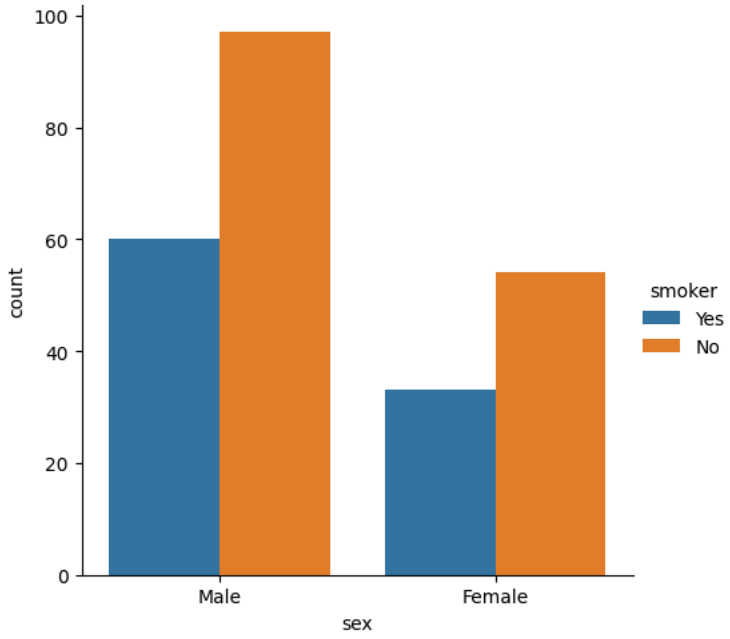

```
#pointplot
sns.catplot(kind='point',data=tips,x='sex',y='total_bill')
plt.show()
```




```
#countplot
sns.catplot(kind='count',data=tips,x='sex')
plt.show()
```




```
#hue
sns.catplot(kind='count',data=tips,x='sex',hue='smoker')
plt.show()
```




Matrix Plot

1.Heatmap

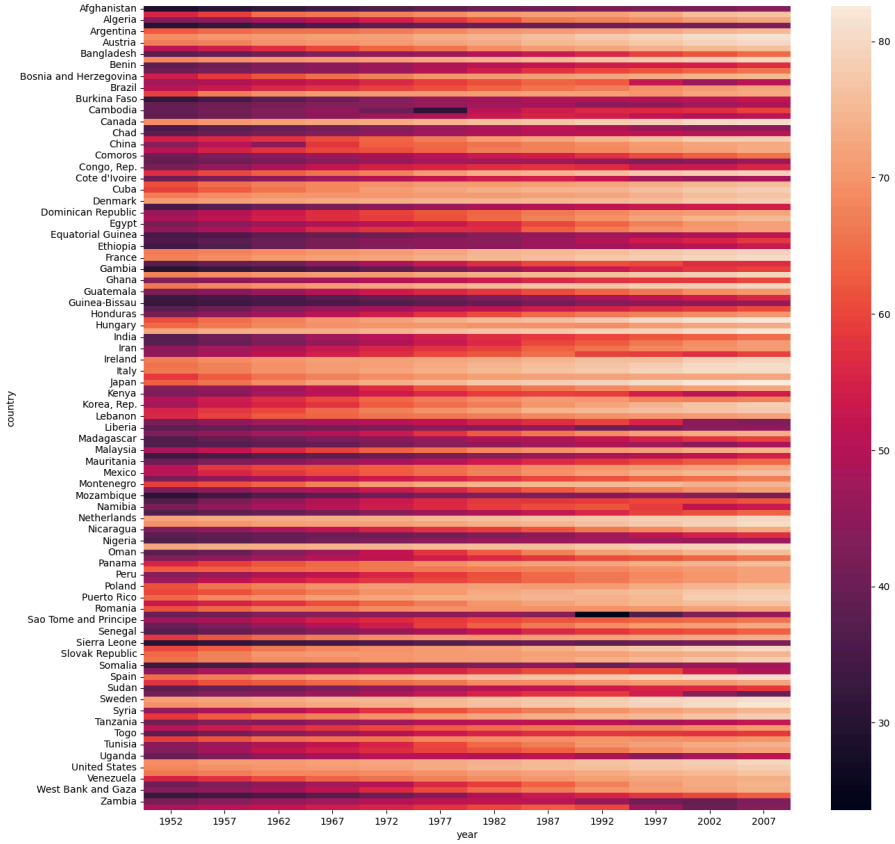
```
gap.head()
```

| | country | continent | year | lifeExp | pop | gdpPercap | iso_alpha | iso_num |
|---|---|---|---|---|---|---|---|---|
| 0 | Afghanistan | Asia | 1952 | 28.801 | 8425333 | 779.445314 | AFG | 4 |
| 1 | Afghanistan | Asia | 1957 | 30.332 | 9240934 | 820.853030 | AFG | 4 |
| 2 | Afghanistan | Asia | 1962 | 31.997 | 10267083 | 853.100710 | AFG | 4 |
| 3 | Afghanistan | Asia | 1967 | 34.020 | 11537966 | 836.197138 | AFG | 4 |
| 4 | Afghanistan | Asia | 1972 | 36.088 | 13079460 | 739.981106 | AFG | 4 |

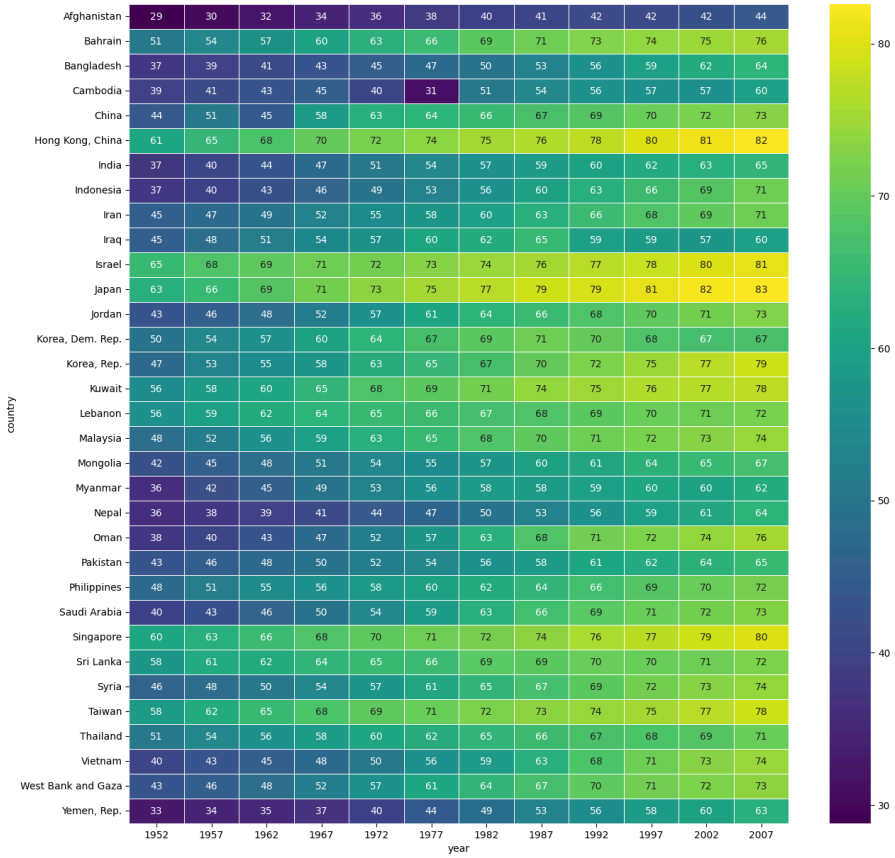
Next steps: Generate code with gap View recommended plots

```
data_heat_map=gap.pivot(index='country',columns='year',values='lifeExp')
```

```
plt.figure(figsize=(15,15))
sns.heatmap(data=data_heat_map)
plt.show()
```




```
#annot
x=gap[gap['continent']=='Asia'].pivot(index='country',columns='year',values='lifeExp')
plt.figure(figsize=(15,15))
sns.heatmap(data=x,annot=True,linewidths=0.5,cmap='viridis')
plt.show()
```




```
#annot
x=gap[gap['continent']=='Asia'].pivot(index='country',columns='year',values='lifeExp')
plt.figure(figsize=(15,15))
sns.heatmap(data=x,annot=True,linewidths=0.5,cmap='summer')
plt.show()
```


2.Clustermap

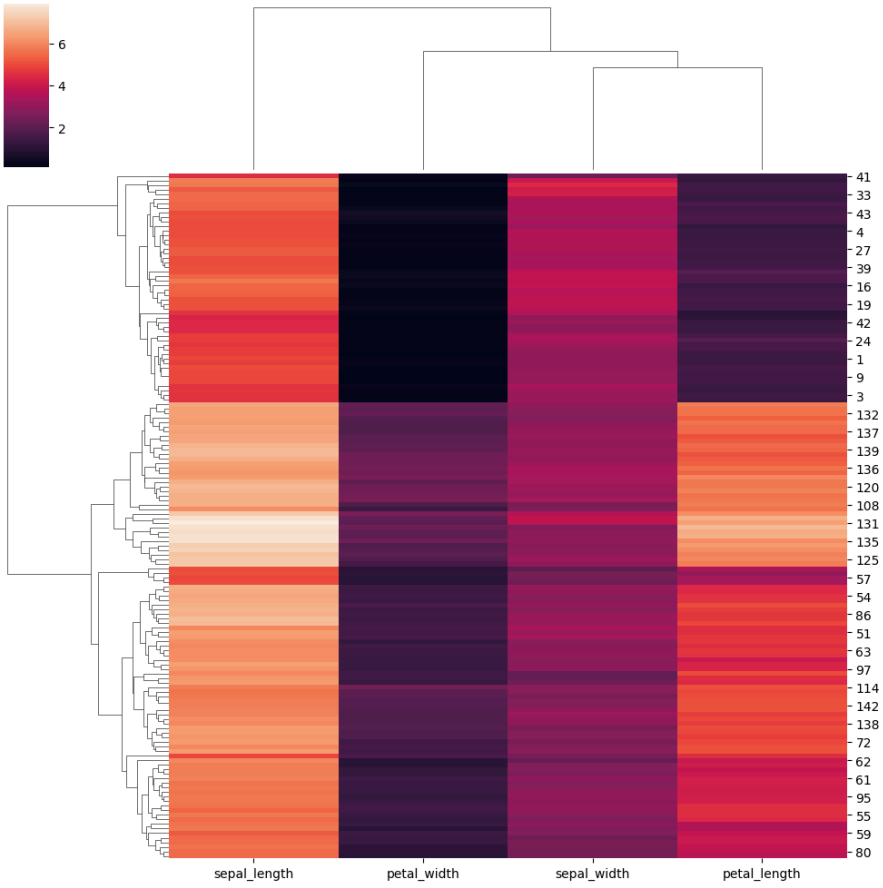
```
iris=px.data.iris()
```

```
iris.head()
```

| | sepal_length | sepal_width | petal_length | petal_width | species | species_id |
|---|---|---|---|---|---|---|
| **0** | 5.1 | 3.5 | 1.4 | 0.2 | setosa | 1 |
| **1** | 4.9 | 3.0 | 1.4 | 0.2 | setosa | 1 |
| **2** | 4.7 | 3.2 | 1.3 | 0.2 | setosa | 1 |
| **3** | 4.6 | 3.1 | 1.5 | 0.2 | setosa | 1 |
| **4** | 5.0 | 3.6 | 1.4 | 0.2 | setosa | 1 |

Next steps: Generate code with `iris` View recommended plots

```
sns.clustermap(iris.iloc[:,[0,1,2,3]])
plt.show()
```




```
Regression plots
```

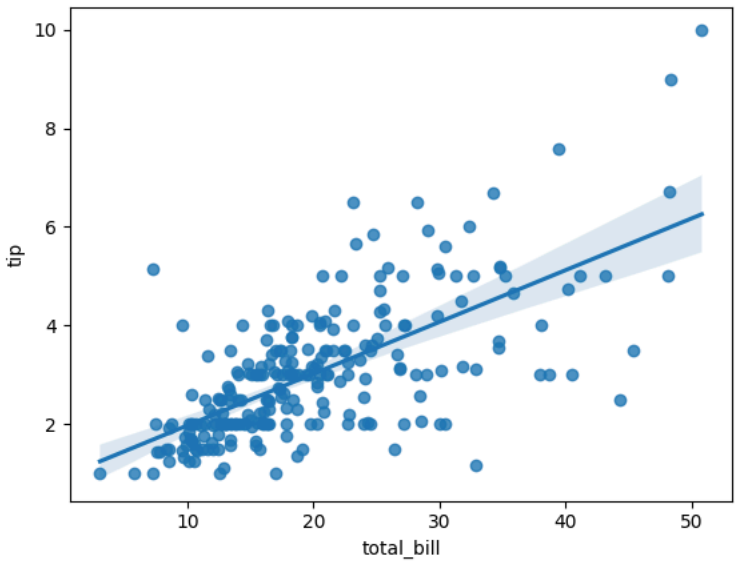
```
tips.head()
```

| | 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 |

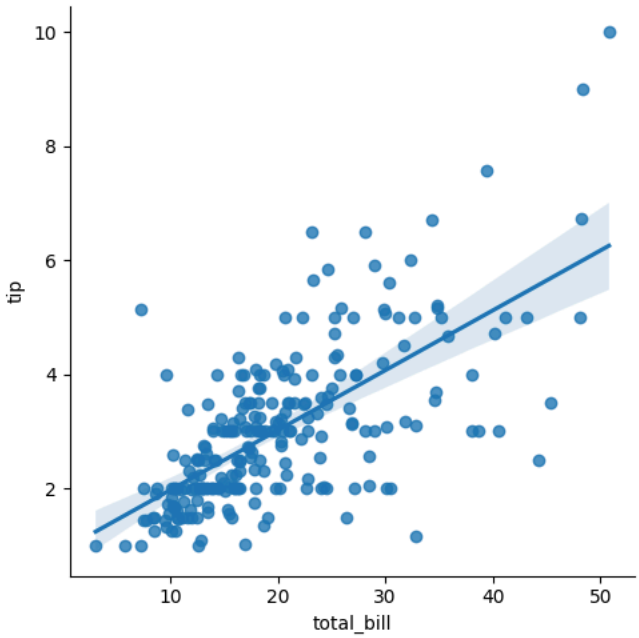
Next steps: Generate code with `tips` View recommended plots

Types: 1.regplot 2.lmplot

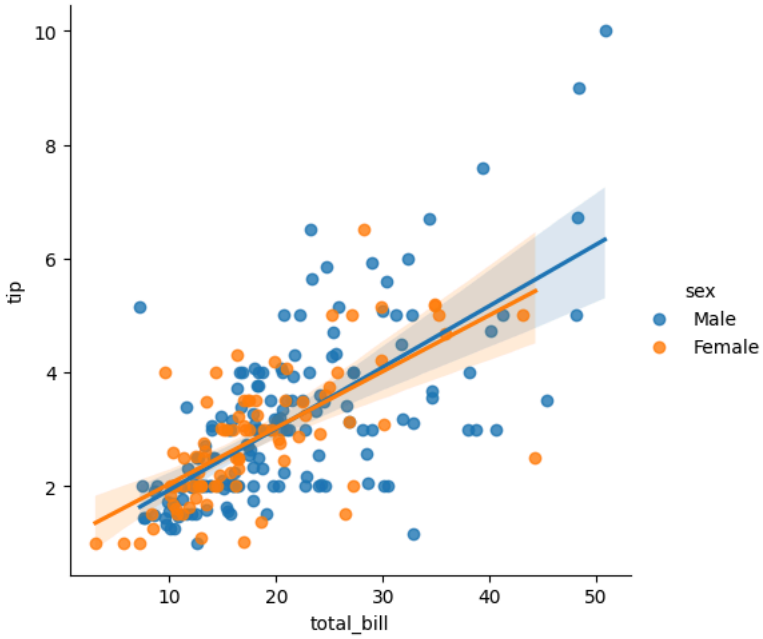
```
#regplot ->axes level
sns.regplot(data=tips,x="total_bill",y="tip")
plt.show()
```



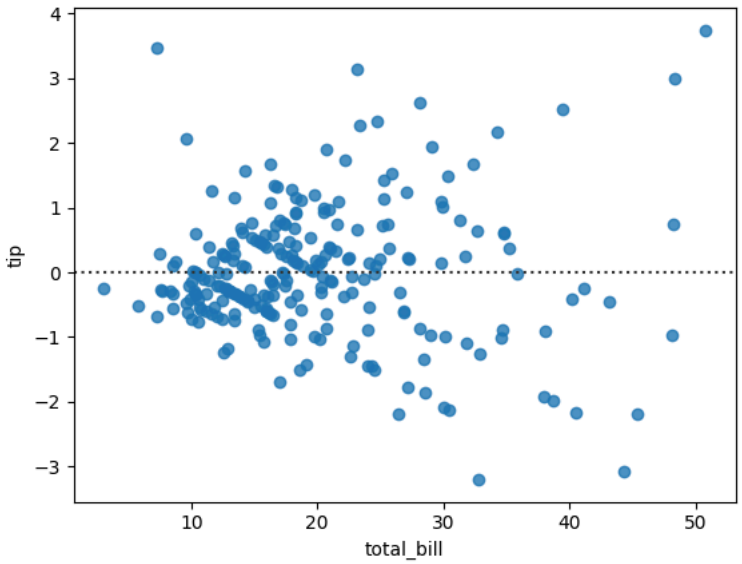

```
#lmplot ->figure level
sns.lmplot(data=tips,x="total_bill",y="tip")
plt.show()
```



```
#hue only present in lmplot
sns.lmplot(data=tips,x="total_bill",y="tip",hue='sex')
plt.show()
```




```
#residplot
sns.residplot(data=tips,x="total_bill",y="tip")
plt.show()
```

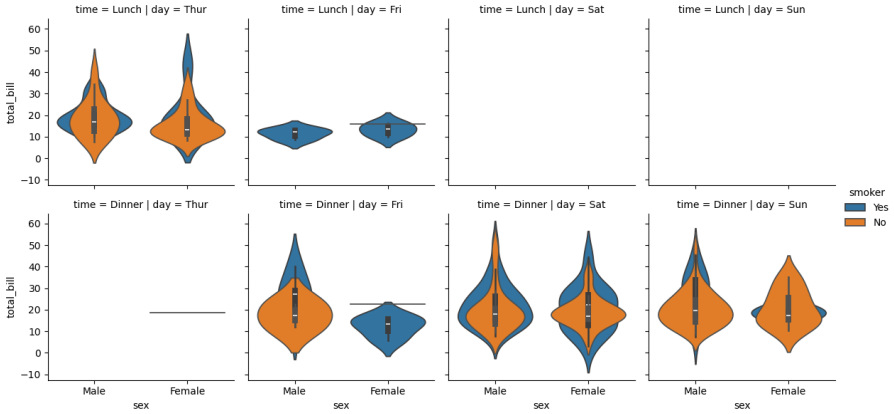



MultiGridplots

FacetGrid

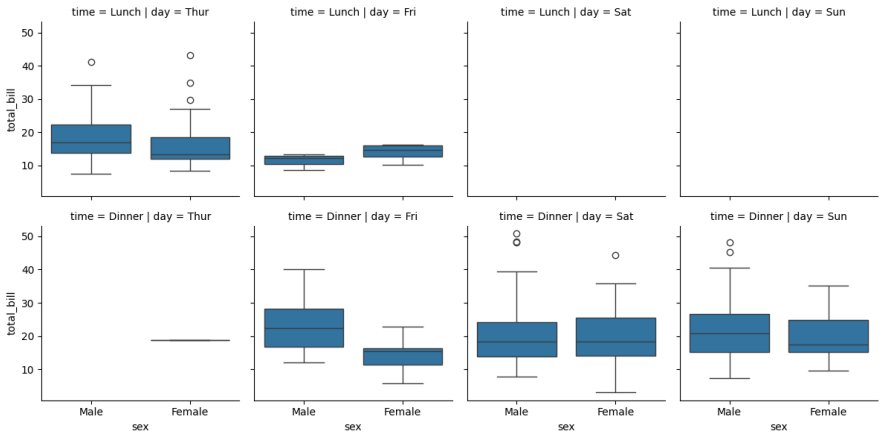
```
g= sns.FacetGrid(data=tips,hue='smoker',col='day',row='time')
g.map(sns.violinplot,'sex','total_bill')
g.add_legend()
```

```
/usr/local/lib/python3.10/dist-packages/seaborn/axisgrid.py:718: UserWarning: Us
  warnings.warn(warning)
<seaborn.axisgrid.FacetGrid at 0x7e88c1534970>
```




```
g= sns.FacetGrid(data=tips,col='day',row='time')
g.map(sns.boxplot,'sex','total_bill')
g.add_legend()
```

```
/usr/local/lib/python3.10/dist-packages/seaborn/axisgrid.py:718: UserWarning: Us
  warnings.warn(warning)
<seaborn.axisgrid.FacetGrid at 0x7e88c1260550>
```




Pairplot

```
iris.head()
```

| | sepal_length | sepal_width | petal_length | petal_width | species | species_id |
|---|---|---|---|---|---|---|
| **0** | 5.1 | 3.5 | 1.4 | 0.2 | setosa | 1 |
| **1** | 4.9 | 3.0 | 1.4 | 0.2 | setosa | 1 |
| **2** | 4.7 | 3.2 | 1.3 | 0.2 | setosa | 1 |
| **3** | 4.6 | 3.1 | 1.5 | 0.2 | setosa | 1 |
| **4** | 5.0 | 3.6 | 1.4 | 0.2 | setosa | 1 |

Next steps: Generate code with `iris` View recommended plots

```
sns.pairplot(data=iris,hue='species')
plt.show()
```