

```
In [1]: ## Data Visualization Libraries  
# Matplotlib and Seaborn
```

## Quantitative Plots

```
In [2]: import seaborn as sns  
import matplotlib.pyplot as plt  
%matplotlib inline
```

```
In [3]: tips = sns.load_dataset('tips')
```

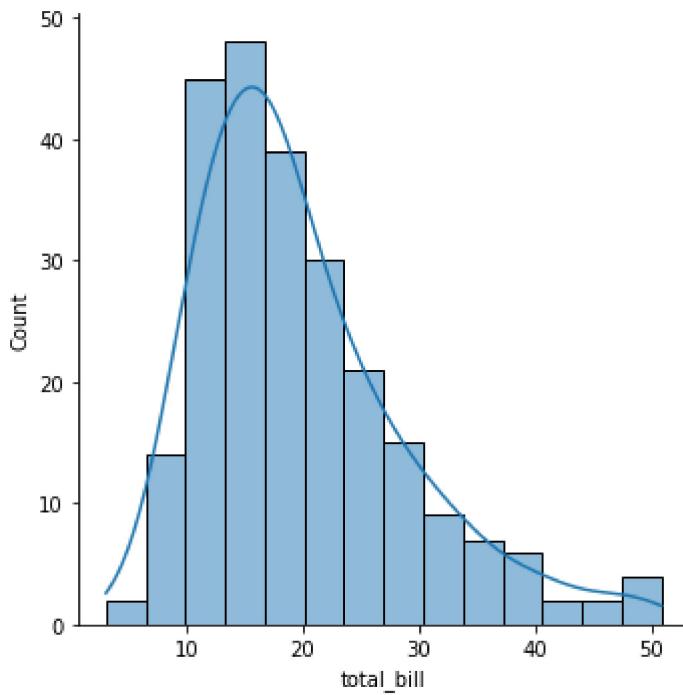
```
In [4]: tips.head()
```

```
Out[4]:
```

	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

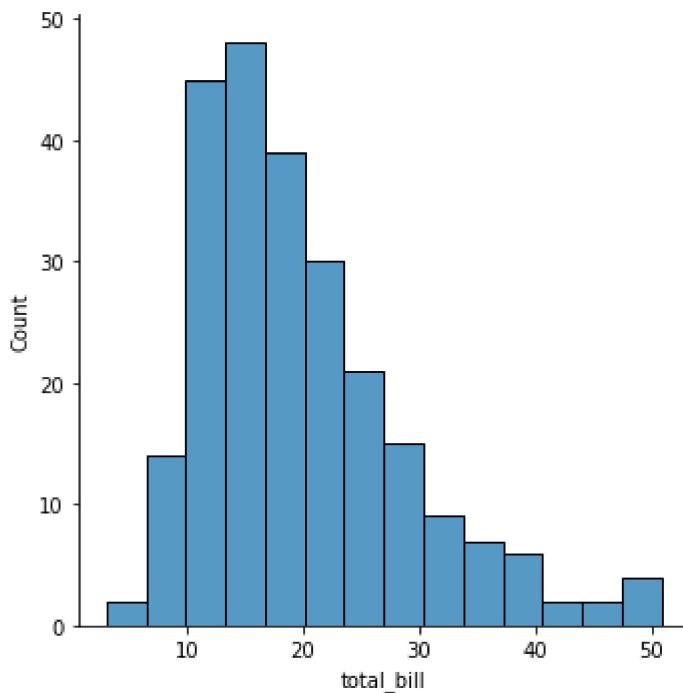
```
In [5]: ## Distribution Plot :  
  
# It is used to check the spread of the data.  
# It is used only for numerical columns  
  
sns.displot(tips.total_bill, kde = True)
```

```
Out[5]: <seaborn.axisgrid.FacetGrid at 0x142929eb430>
```



```
In [6]: sns.displot(tips.total_bill, kde = False)
```

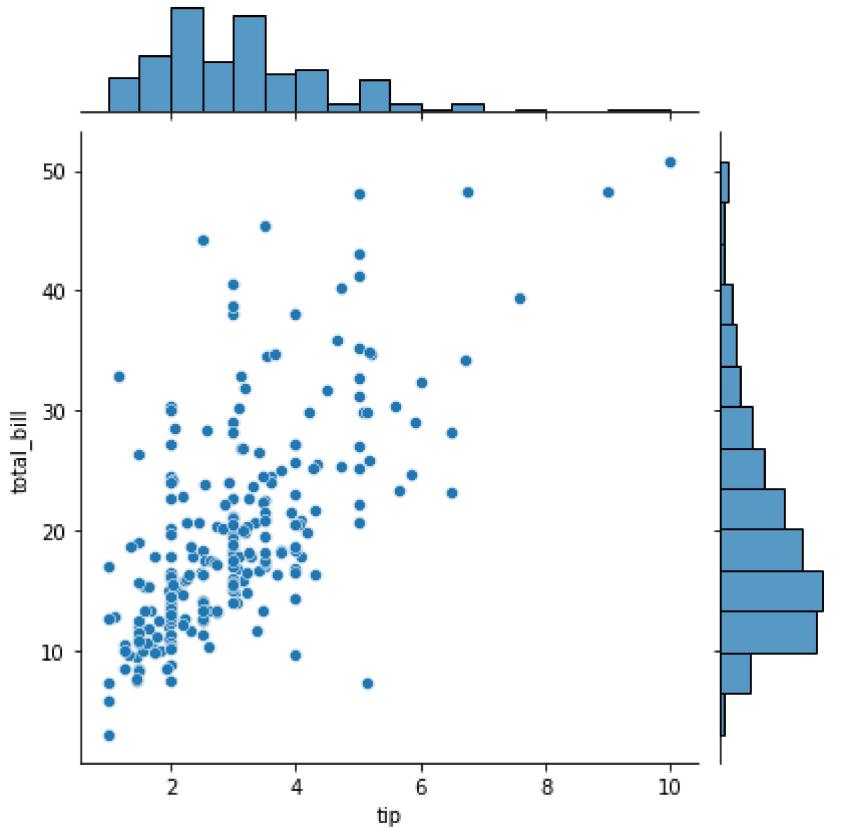
```
Out[6]: <seaborn.axisgrid.FacetGrid at 0x14293186be0>
```



```
In [7]: ## Joint Plot :- To check the correlation between variables
```

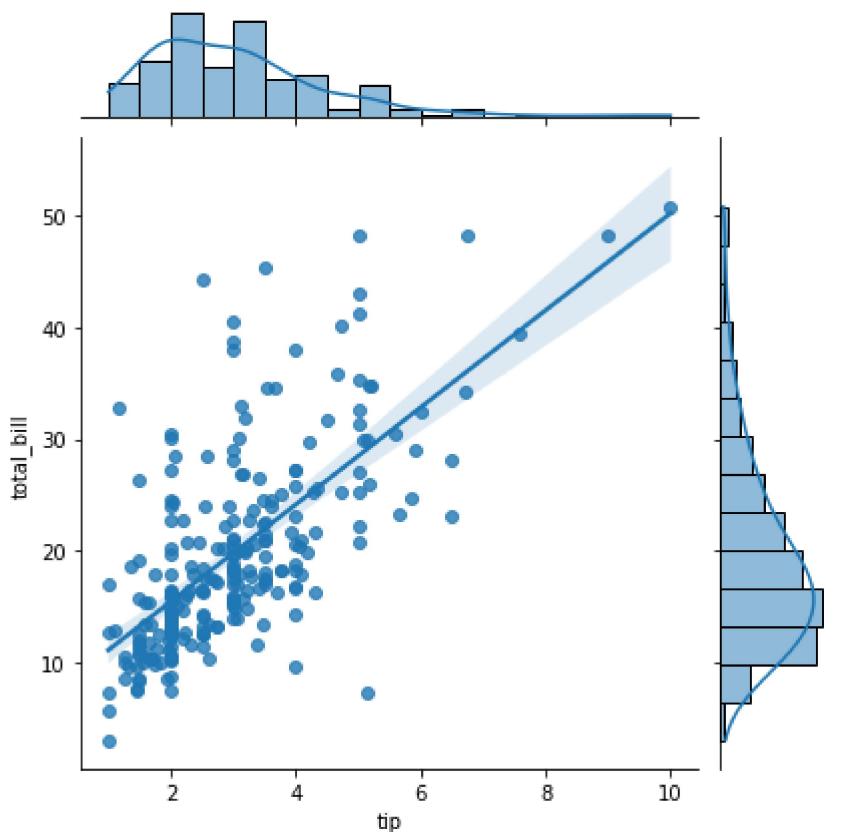
```
sns.jointplot(x = tips.tip, y = tips.total_bill )
```

```
Out[7]: <seaborn.axisgrid.JointGrid at 0x1428ce065b0>
```



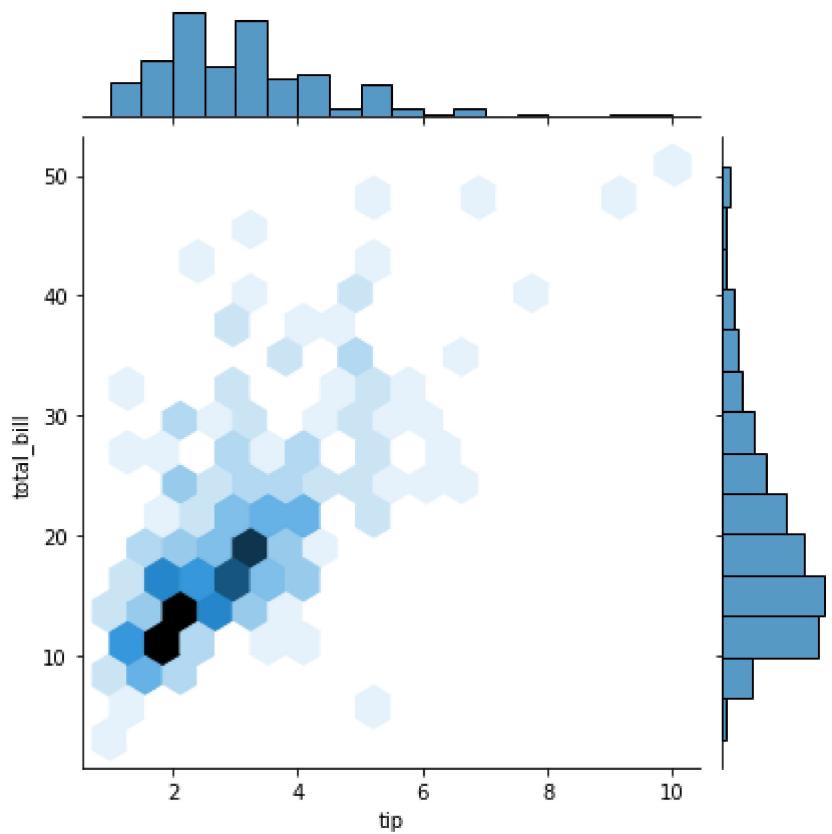
```
In [8]: sns.jointplot(x = tips.tip, y=tips.total_bill, kind = "reg")
```

```
Out[8]: <seaborn.axisgrid.JointGrid at 0x142937b2eb0>
```



```
In [9]: sns.jointplot(x = tips.tip, y=tips.total_bill, kind = "hex" )
```

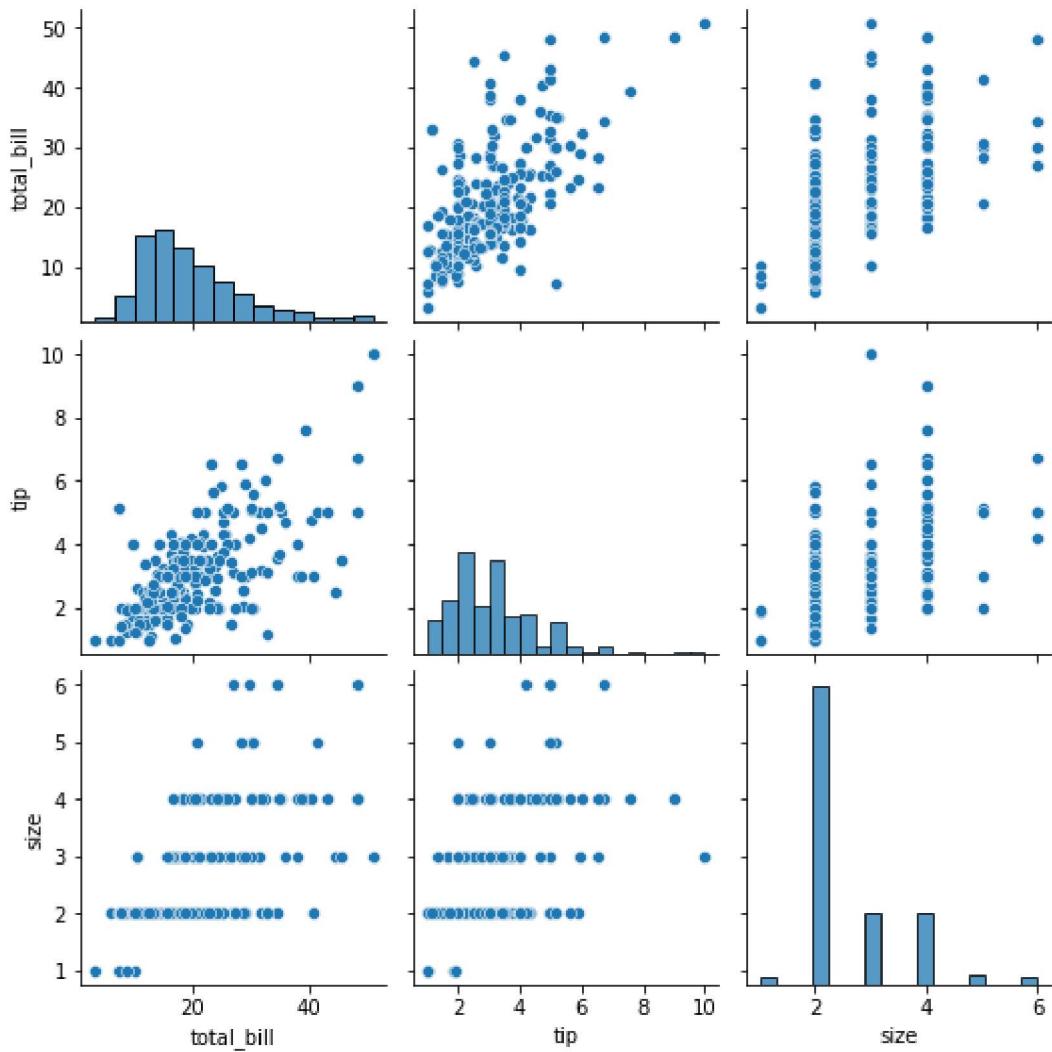
```
Out[9]: <seaborn.axisgrid.JointGrid at 0x1429396d6d0>
```



```
In [10]: ## Pairplot Used for Numerical Columns
```

```
sns.pairplot(tips)
```

```
Out[10]: <seaborn.axisgrid.PairGrid at 0x14294a9b760>
```

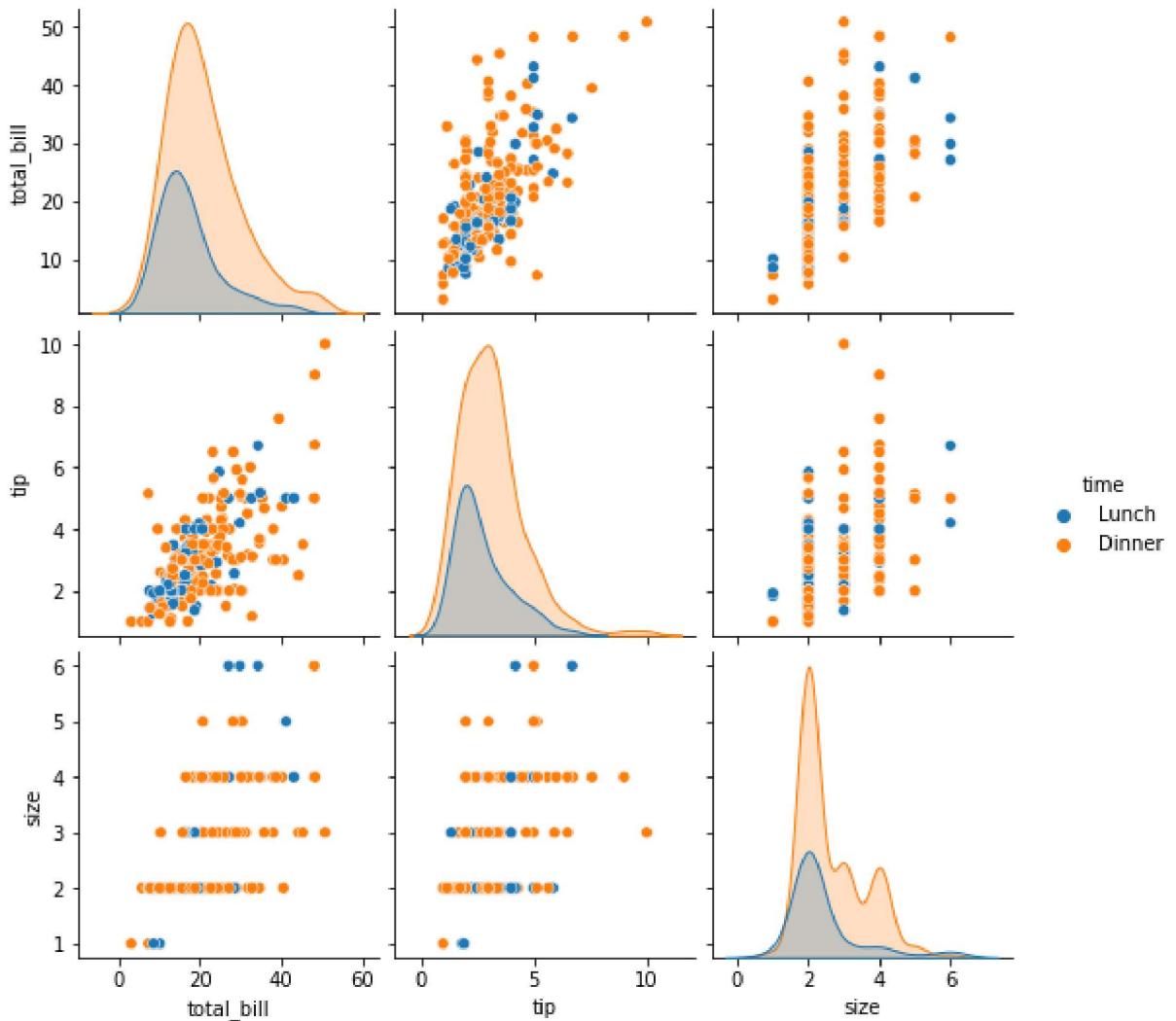


```
In [11]: tips.time.value_counts()
```

```
Out[11]: Dinner    176  
Lunch      68  
Name: time, dtype: int64
```

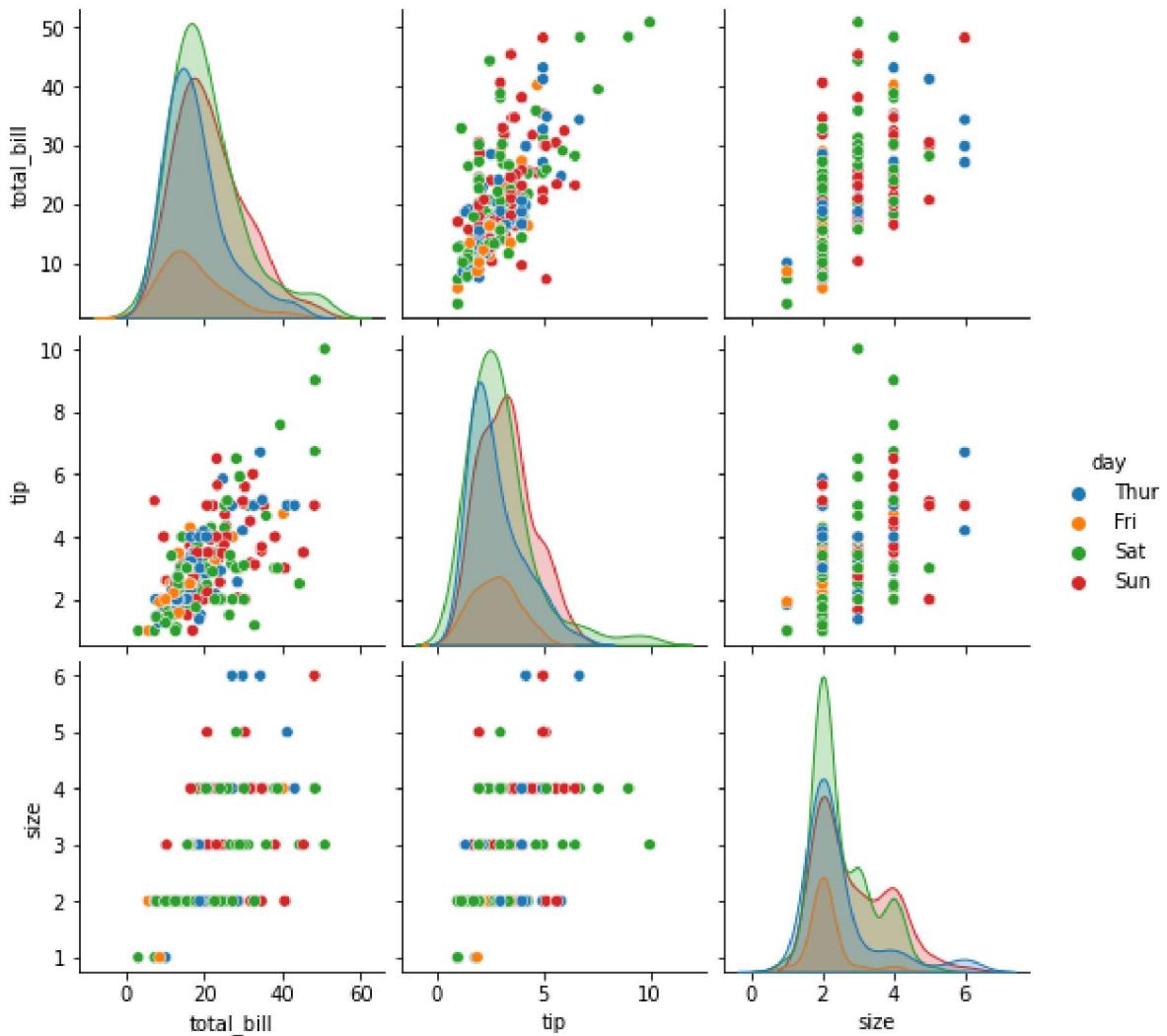
```
In [12]: sns.pairplot(tips , hue='time')
```

```
Out[12]: <seaborn.axisgrid.PairGrid at 0x14292a9fc40>
```



```
In [13]: sns.pairplot(tips, hue = 'day')
```

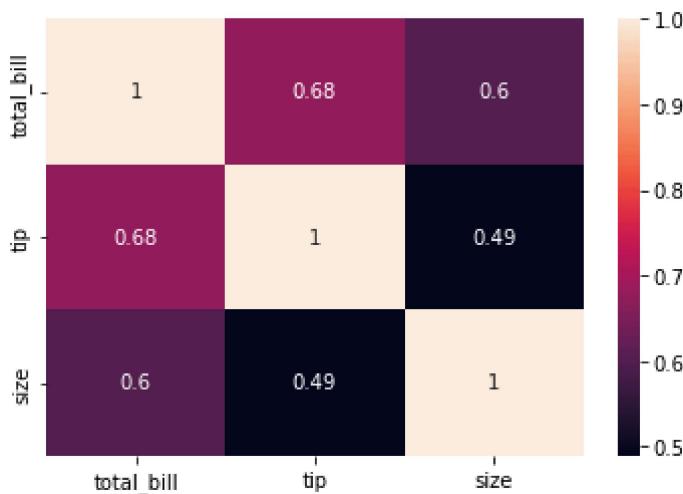
```
Out[13]: <seaborn.axisgrid.PairGrid at 0x142956090a0>
```



```
In [14]: #Correlation Analysis
#Heatmap
```

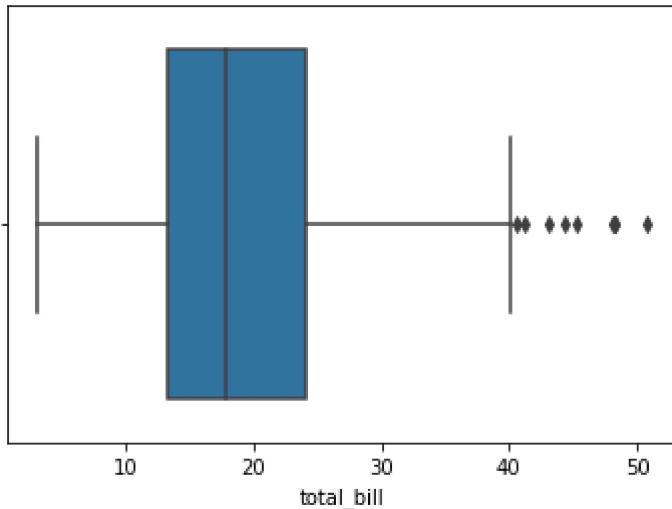
```
sns.heatmap(tips.corr(), annot=True)
```

```
Out[14]: <AxesSubplot:>
```



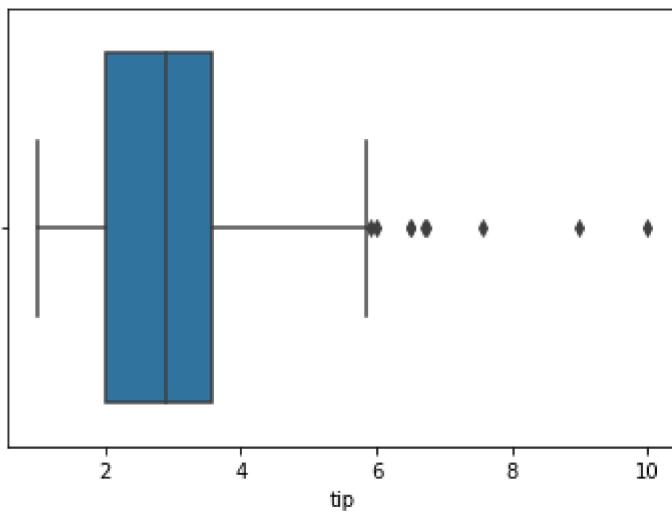
```
In [15]: #Box Plot
sns.boxplot(tips.total_bill)
```

```
C:\Users\SW20407278\Anaconda3\lib\site-packages\seaborn\_decorators.py:36: FutureWarning: Pass the following variable as a keyword arg: x. From version 0.12, the only valid positional argument will be `data`, and passing other arguments without an explicit keyword will result in an error or misinterpretation.  
    warnings.warn(  
Out[15]: <AxesSubplot:xlabel='total_bill'>
```



```
In [16]: sns.boxplot(tips.tip)
```

```
C:\Users\SW20407278\Anaconda3\lib\site-packages\seaborn\_decorators.py:36: FutureWarning: Pass the following variable as a keyword arg: x. From version 0.12, the only valid positional argument will be `data`, and passing other arguments without an explicit keyword will result in an error or misinterpretation.  
    warnings.warn(  
Out[16]: <AxesSubplot:xlabel='tip'>
```

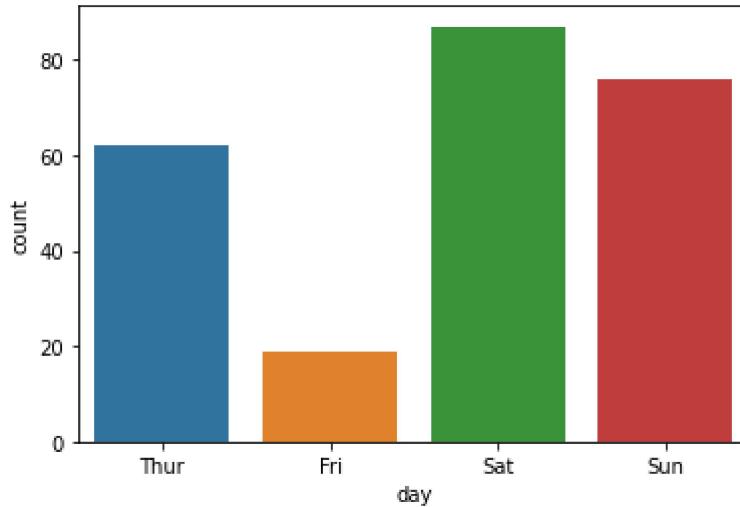


## Qualitative Plots

```
In [ ]: ## For Categorical data, we use  
## bar chart, pie-chart, count plot
```

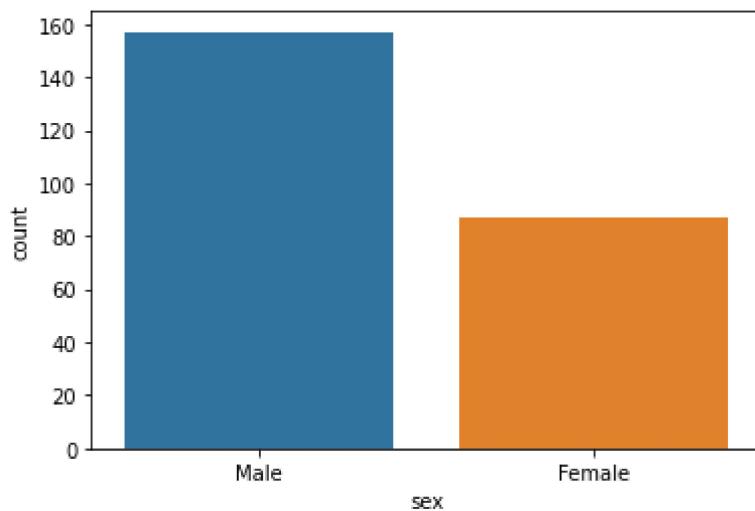
```
In [18]: sns.countplot(tips.day)
```

```
C:\Users\SW20407278\Anaconda3\lib\site-packages\seaborn\_decorators.py:36: FutureWarning: Pass the following variable as a keyword arg: x. From version 0.12, the only valid positional argument will be `data`, and passing other arguments without an explicit keyword will result in an error or misinterpretation.  
    warnings.warn(  
Out[18]: <AxesSubplot:xlabel='day', ylabel='count'>
```



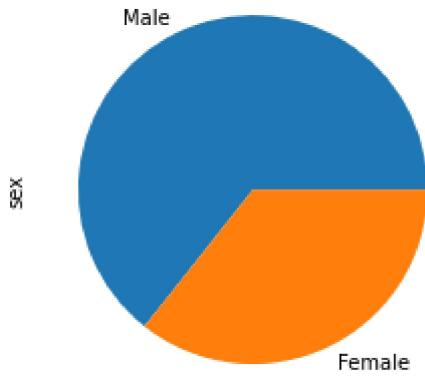
```
In [19]: sns.countplot(tips.sex)
```

```
C:\Users\SW20407278\Anaconda3\lib\site-packages\seaborn\_decorators.py:36: FutureWarning: Pass the following variable as a keyword arg: x. From version 0.12, the only valid positional argument will be `data`, and passing other arguments without an explicit keyword will result in an error or misinterpretation.  
    warnings.warn(  
Out[19]: <AxesSubplot:xlabel='sex', ylabel='count'>
```



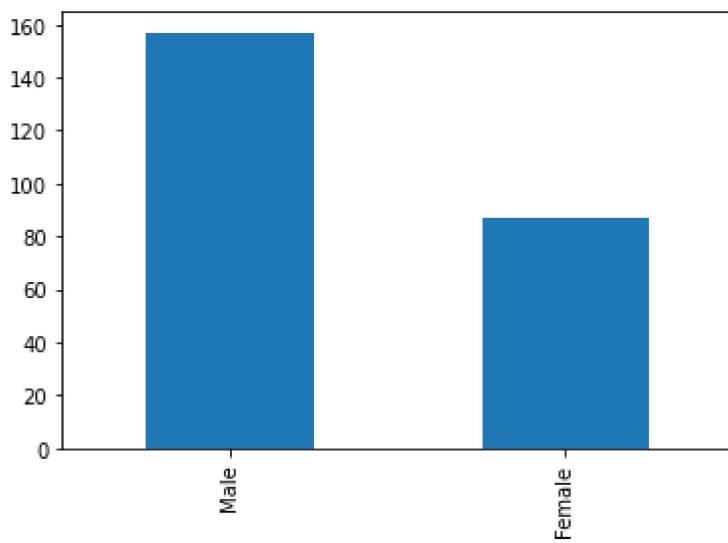
```
In [20]: tips.sex.value_counts().plot(kind = 'pie')
```

```
Out[20]: <AxesSubplot:ylabel='sex'>
```



```
In [21]: tips.sex.value_counts().plot(kind= 'bar')
```

```
Out[21]: <AxesSubplot:>
```

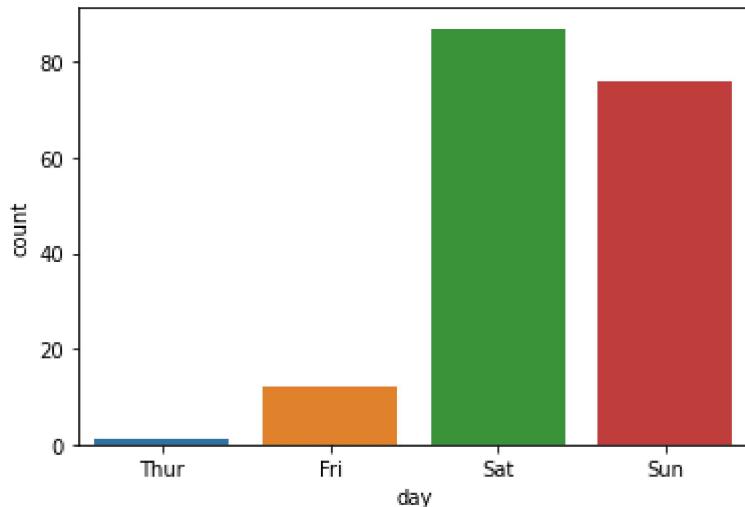


```
In [22]: sns.countplot(tips[tips.time == 'Dinner']['day'])
```

```
C:\Users\SW20407278\Anaconda3\lib\site-packages\seaborn\_decorators.py:36: FutureWarning: Pass the following variable as a keyword arg: x. From version 0.12, the only valid positional argument will be `data`, and passing other arguments without an explicit keyword will result in an error or misinterpretation.
```

```
    warnings.warn(
```

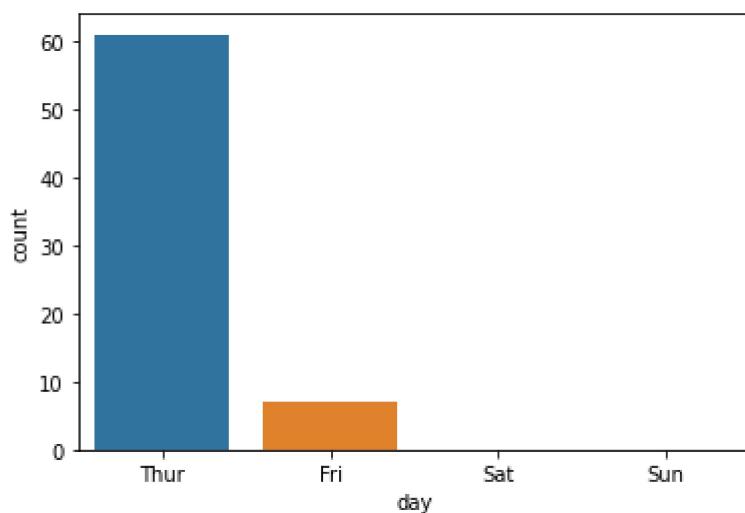
```
Out[22]: <AxesSubplot:xlabel='day', ylabel='count'>
```



```
In [23]: sns.countplot(tips[tips.time == 'Lunch'][day])
```

```
C:\Users\SW20407278\Anaconda3\lib\site-packages\seaborn\_decorators.py:36: FutureWarning: Pass the following variable as a keyword arg: x. From version 0.12, the only valid positional argument will be `data`, and passing other arguments without an explicit keyword will result in an error or misinterpretation.  
    warnings.warn(
```

```
Out[23]: <AxesSubplot:xlabel='day', ylabel='count'>
```



```
In [24]: ### Grid Graph which will allow you to map different plots to grid rows and columns.
```

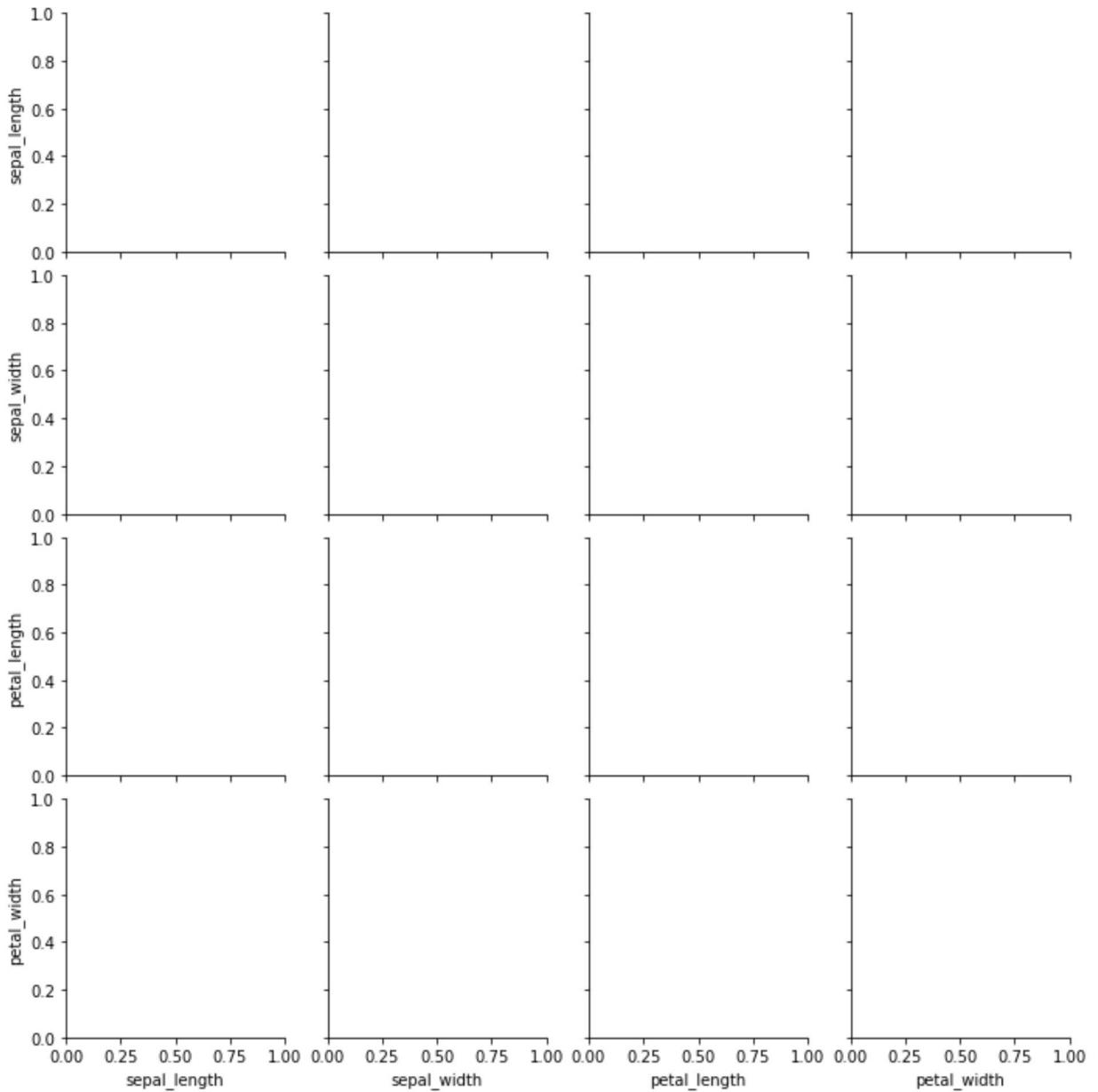
```
In [25]: iris = sns.load_dataset('iris')
```

```
In [26]: iris.info()
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 150 entries, 0 to 149
Data columns (total 5 columns):
 #   Column      Non-Null Count  Dtype  
--- 
 0   sepal_length  150 non-null    float64 
 1   sepal_width   150 non-null    float64 
 2   petal_length  150 non-null    float64 
 3   petal_width   150 non-null    float64 
 4   species       150 non-null    object  
dtypes: float64(4), object(1)
memory usage: 6.0+ KB
```

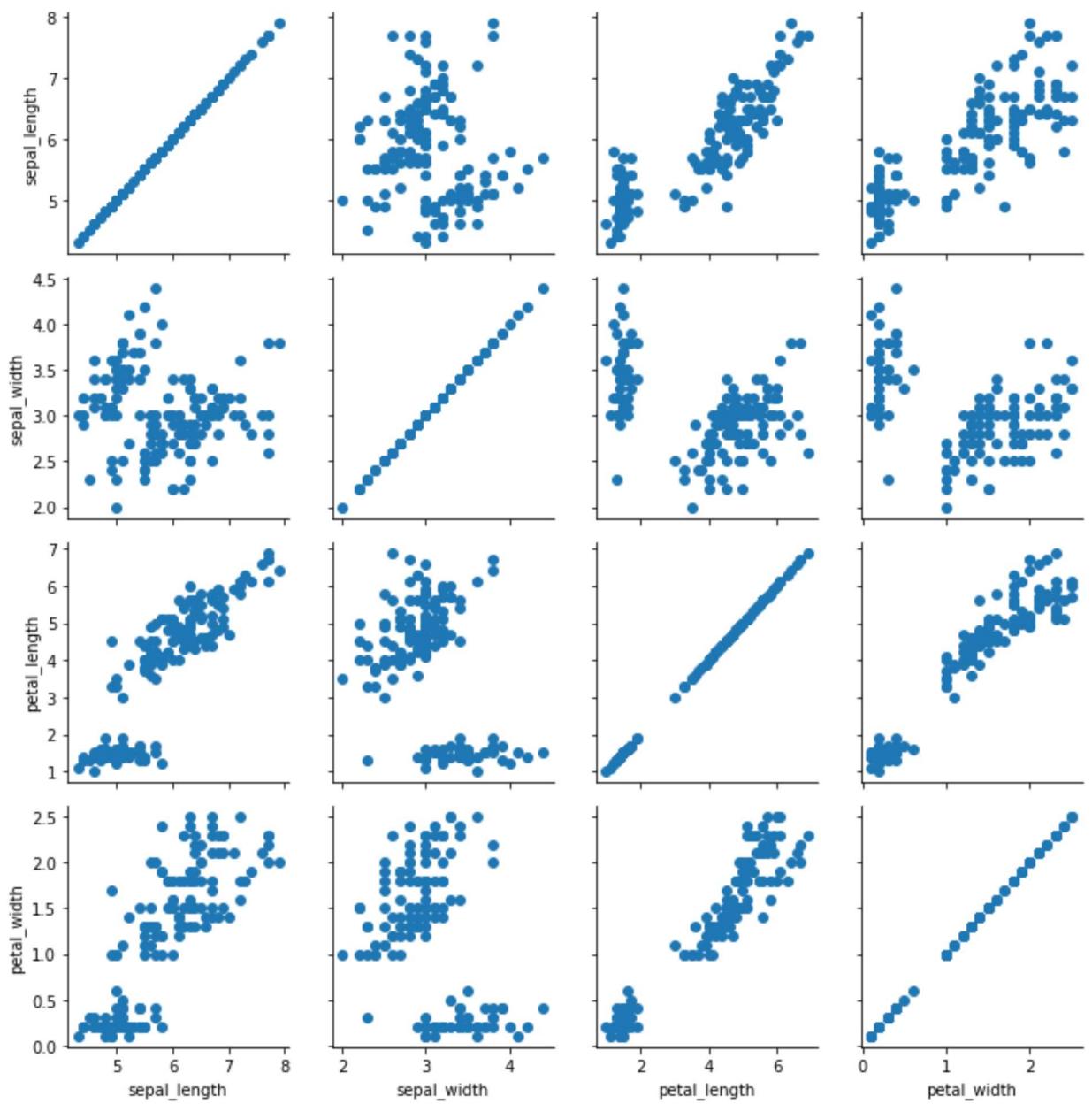
```
In [27]: sns.PairGrid(iris)
```

```
Out[27]: <seaborn.axisgrid.PairGrid at 0x142972d8040>
```



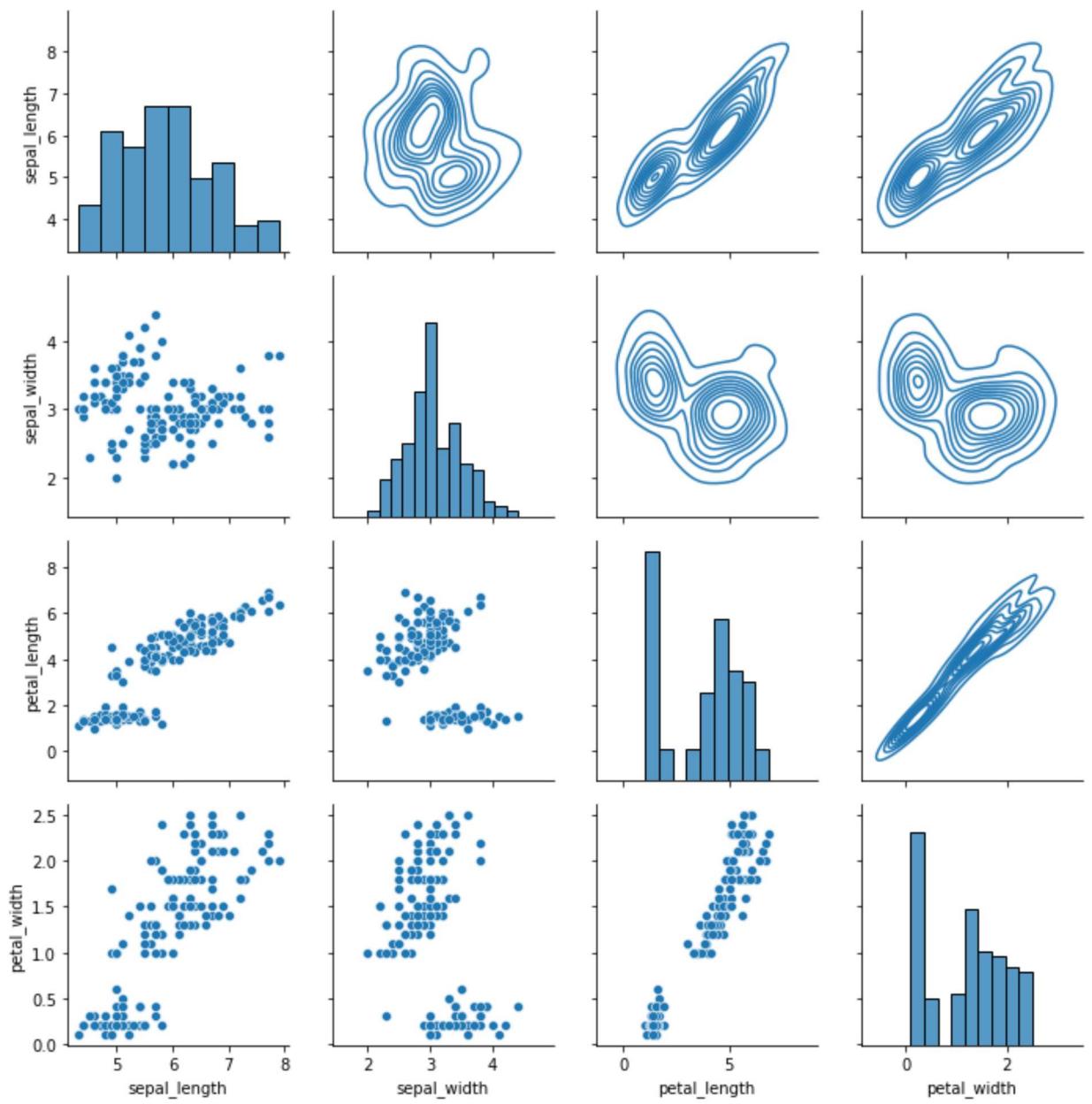
```
In [29]: po = sns.PairGrid(iris)
po.map(plt.scatter)
```

```
Out[29]: <seaborn.axisgrid.PairGrid at 0x1429720b2e0>
```



```
In [33]: po = sns.PairGrid(iris)
po.map_diag(sns.histplot)
po.map_lower(sns.scatterplot)
po.map_upper(sns.kdeplot)
```

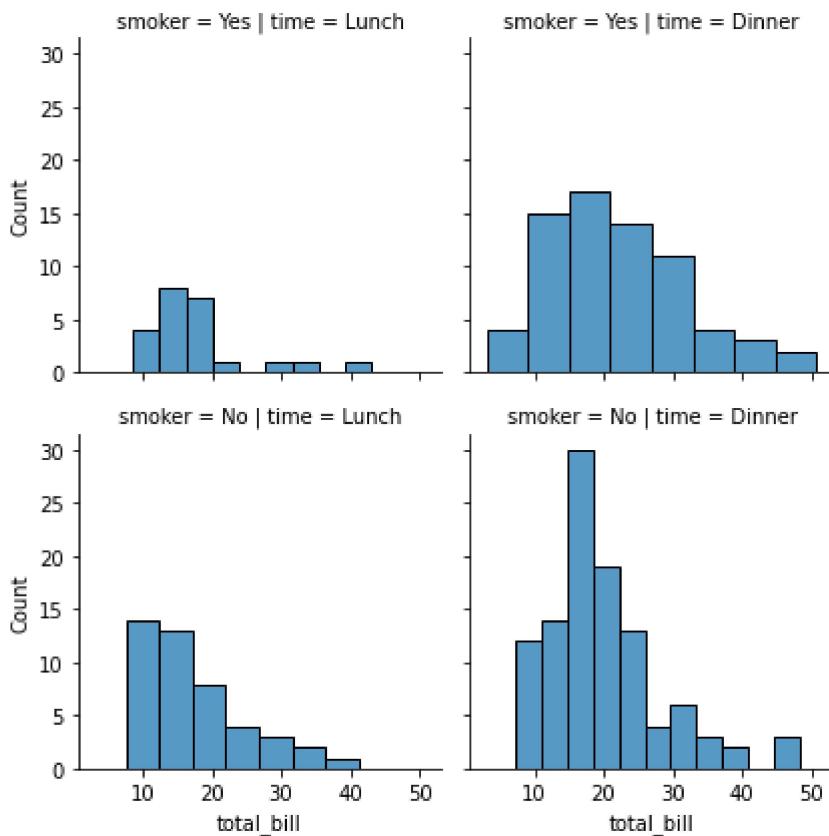
```
Out[33]: <seaborn.axisgrid.PairGrid at 0x1429806d520>
```



```
In [ ]: ### Facet Grid
```

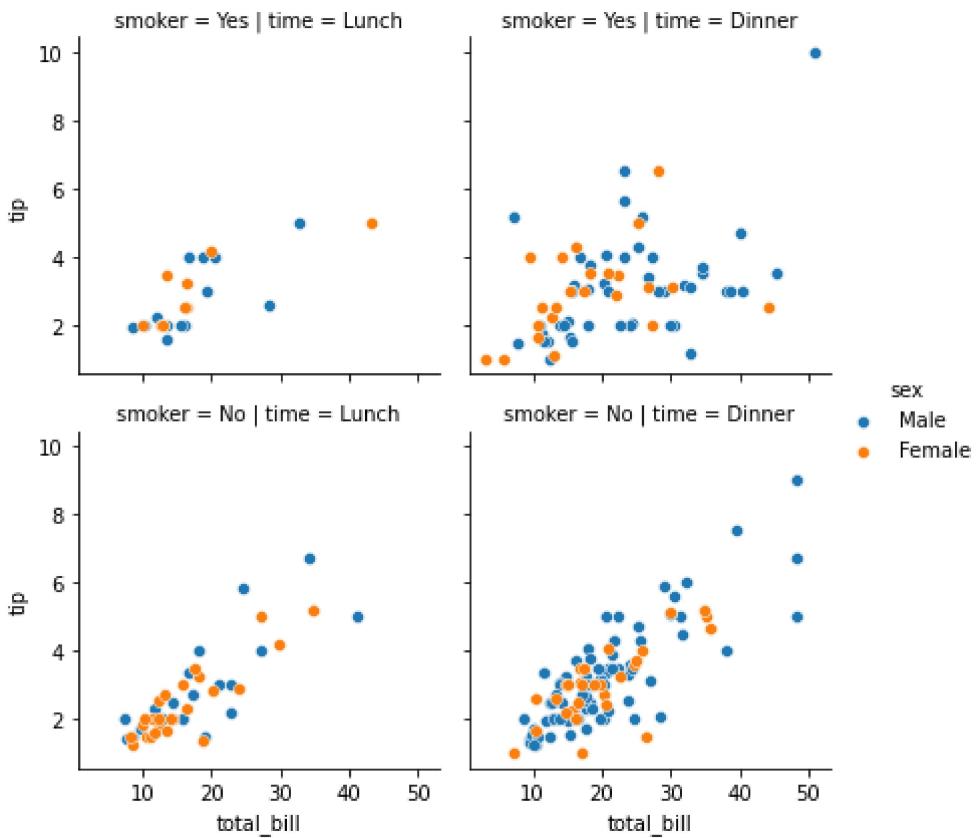
```
In [39]: fg =sns.FacetGrid(tips, row = 'smoker', col = 'time')
fg.map(sns.histplot, "total_bill")
```

```
Out[39]: <seaborn.axisgrid.FacetGrid at 0x1429d5a20a0>
```



```
In [41]: fg = sns.FacetGrid(tips, row = "smoker", col = "time", hue = "sex")
fg.map(sns.scatterplot, "total_bill", "tip").add_legend()
```

```
Out[41]: <seaborn.axisgrid.FacetGrid at 0x1429d2da310>
```



```
In [ ]:
```