# MATPLOTLIB

```
import numpy as n #importing numpy library
x=n.linspace(0,5,11)
y=x ** 2
```

```
x #for plotting as x vales in graph
```

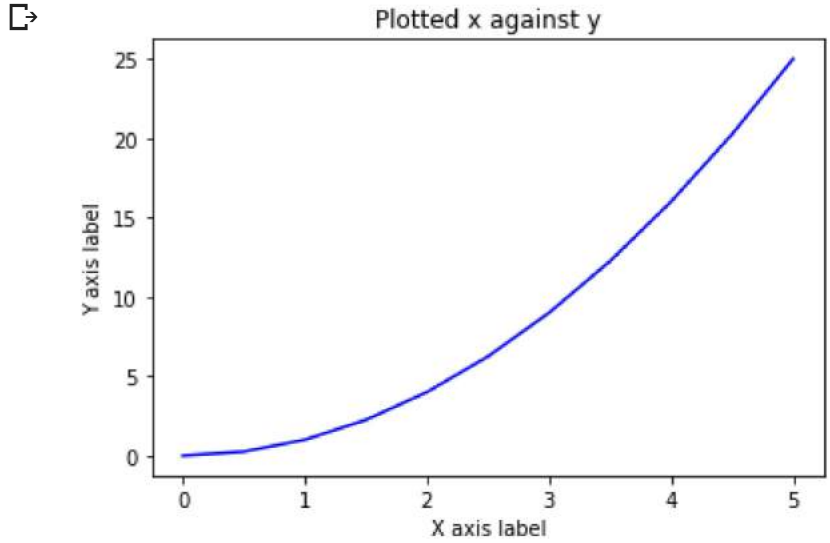
```
array([0. , 0.5, 1. , 1.5, 2. , 2.5, 3. , 3.5, 4. , 4.5, 5. ])
```

```
y #for plotting as y vales in graph
```

```
array([ 0.  ,  0.25,  1.  ,  2.25,  4.  ,  6.25,  9.  , 12.25, 16.  ,
       20.25, 25.  ])
```

```
import matplotlib .pyplot as p #importing matplotlib library
```

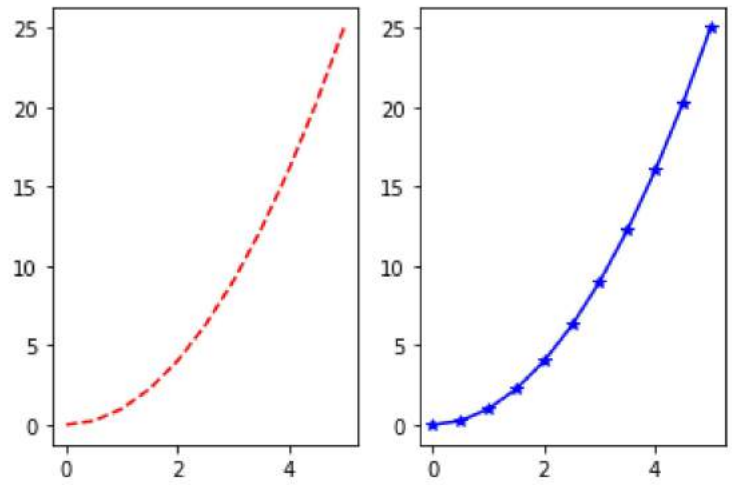
```
p.plot(x,y,'b') #'b' is color blue
p.xlabel('X axis label') #like number,frequency
p.ylabel('Y axis label') #like time..
p.title('Plotted x against y')
p.show()
```




## Using multiple plots

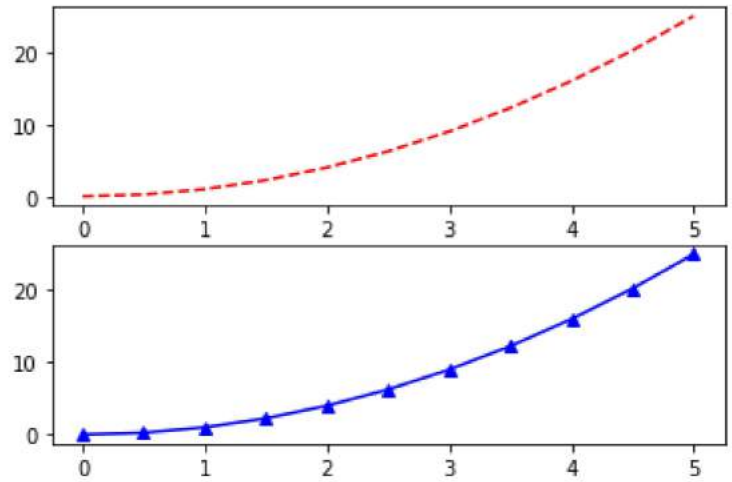
```
p.subplot(1,2,1)                #(no of rows in the canvas. no of cols in the same canvas,fi
p.plot(x,y,'r--')               # 1st row ile 1st col for graph no:1
p.subplot(1,2,2)
p.plot(x,y,"b*-")               # 1st row ile 2nd col for graph no:2
```

```
[<matplotlib.lines.Line2D at 0x7f6adf7d3190>]
```



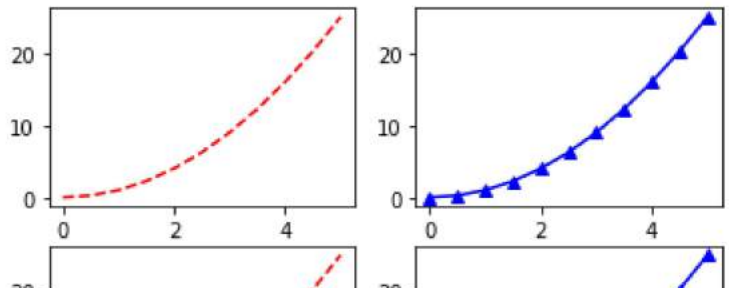
```
p.subplot(2,1,1)
p.plot(x,y,'r--')
p.subplot(2,1,2)
p.plot(x,y,"b^-")
```

```
[<matplotlib.lines.Line2D at 0x7f6adef82450>]
```



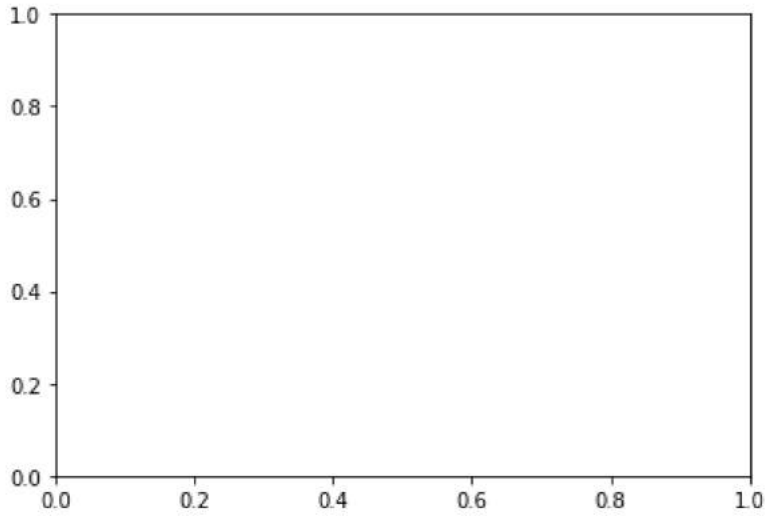
```
p.subplot(2,2,1)
p.plot(x,y,'r--')
p.subplot(2,2,2)
p.plot(x,y,"b^-")
p.subplot(2,2,3)
p.plot(x,y,'r--')
p.subplot(2,2,4)
p.plot(x,y,"b^-")
```

```
[<matplotlib.lines.Line2D at 0x7f6adee39890>]
```



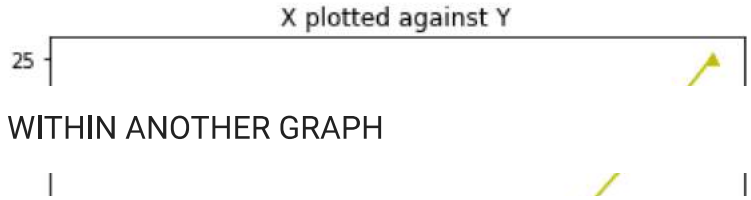

## MATPLOTLIB USING OOP's

```
f=p.figure()
axes=f.add_axes([0.5,0.5,0.8,0.8])  #(left,bottom,height,width)
```




```
f=p.figure()
axes=f.add_axes([0.5,0.5,0.8,0.8])
axes.set_xlabel('X label')
axes.set_ylabel('Y label')
axes.set_title('X plotted against Y')
axes.plot(x,y,"y^-")
```

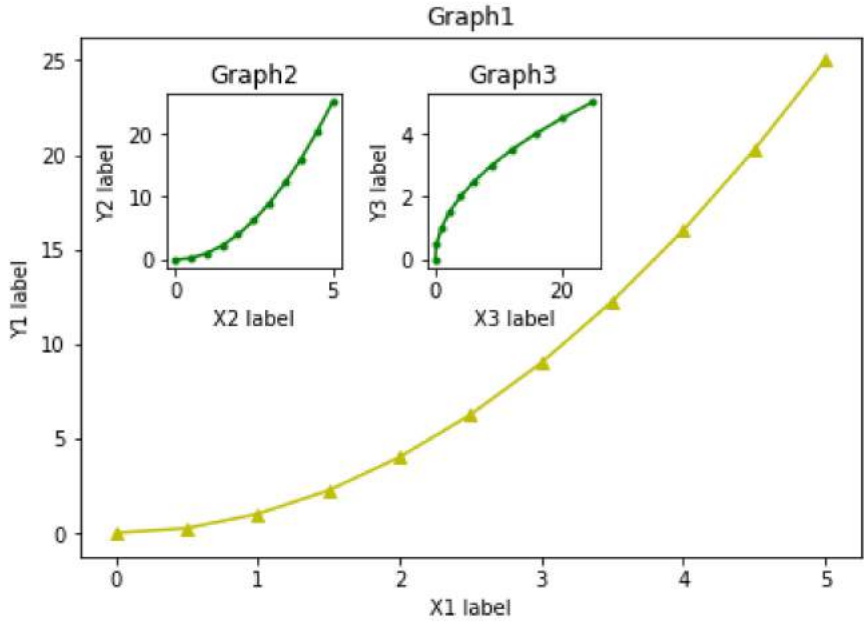
```
[<matplotlib.lines.Line2D at 0x7f6ada7323d0>]
```




## GRAPH WITHIN ANOTHER GRAPH

```
f=p.figure()
axes1=f.add_axes([0.5,0.5,0.9,0.9])
axes1.set_xlabel('X1 label')
axes1.set_ylabel('Y1 label')
axes1.set_title('Graph1')
axes1.plot(x,y,"y^-")

axes2=f.add_axes([0.6,1,0.2,0.3])   #to make the 2nd  graph come within the 1st, change the f
axes2.set_xlabel('X2 label')
axes2.set_ylabel('Y2 label')
axes2.set_title('Graph2')
axes2.plot(x,y,"g.-")

axes3=f.add_axes([0.9,1,0.2,0.3])   #to make the 2nd  graph come within the 1st, change the f
axes3.set_xlabel('X3 label')
axes3.set_ylabel('Y3 label')
axes3.set_title('Graph3')
axes3.plot(y,x,"g.-") #when plot(y,x) is given the graph becomes inverse(opp direction)
```

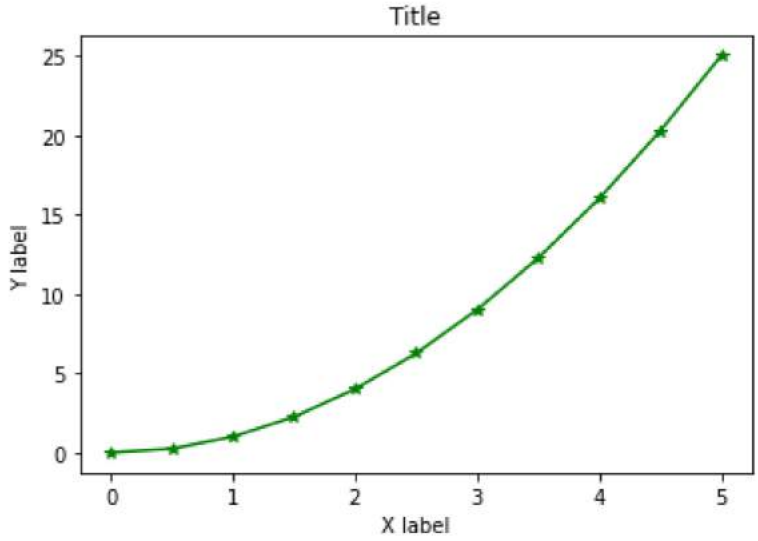
```
[<matplotlib.lines.Line2D at 0x7f6ad9a46b50>]
```




```
f,axes=p.subplots()

axes.plot(x,y,'g*-')
axes.set_xlabel("X label")
```

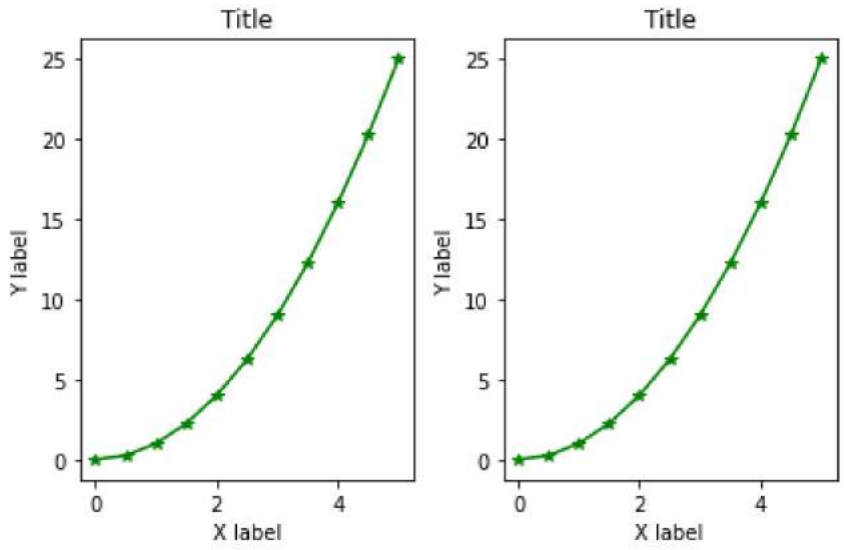
```
axes.set_ylabel("Y label")
axes.set_title("Title")
```

```
Text(0.5, 1.0, 'Title')
```




## CONCEPT OF SUBPLOTS IN OOP's

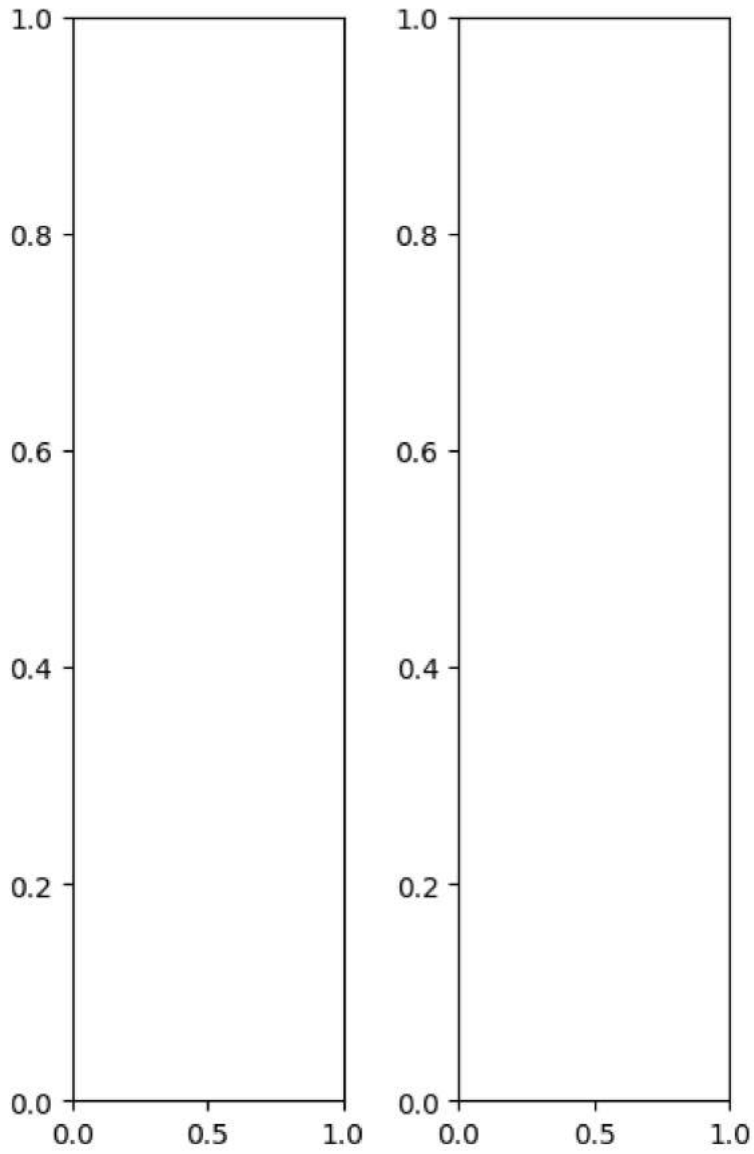
```
fig,axes=p.subplots(1,2)

for ax in axes:
  ax.plot(x,y,'g*-')
  ax.set_xlabel("X label")
  ax.set_ylabel("Y label")
  ax.set_title("Title")
  fig.tight_layout()
```



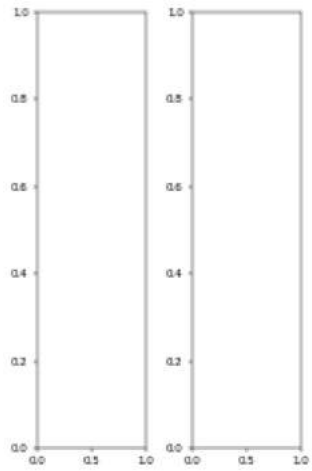

## CONTROLLING FIGURE SIZE AND DPI

```
fig,axes=p.subplots(1,2,figsize=(4,6),dpi=100)  #figsize(width,height)
```

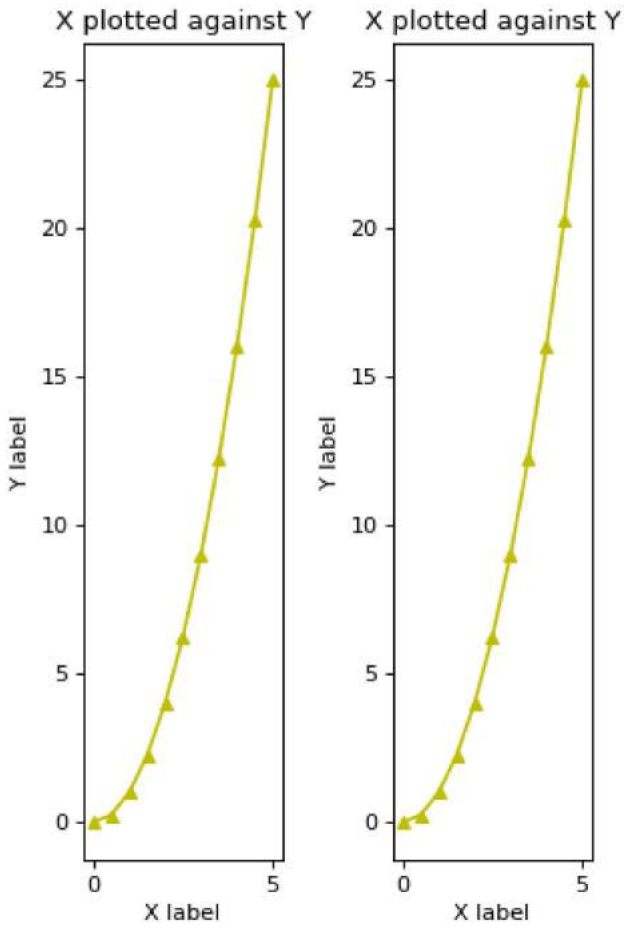
```
fig.tight_layout()
```




```
fig,axes=p.subplots(1,2,figsize=(4,6),dpi=40)  #figsize(width,height)
fig.tight_layout()
```



```
fig,axes=p.subplots(1,2,figsize=(4,6),dpi=80)  #figsize(width,height)
for axes in axes:
  axes.set_xlabel('X label')
  axes.set_ylabel('Y label')
  axes.set_title('X plotted against Y')
  axes.plot(x,y,"y^-")
fig.tight_layout()
```



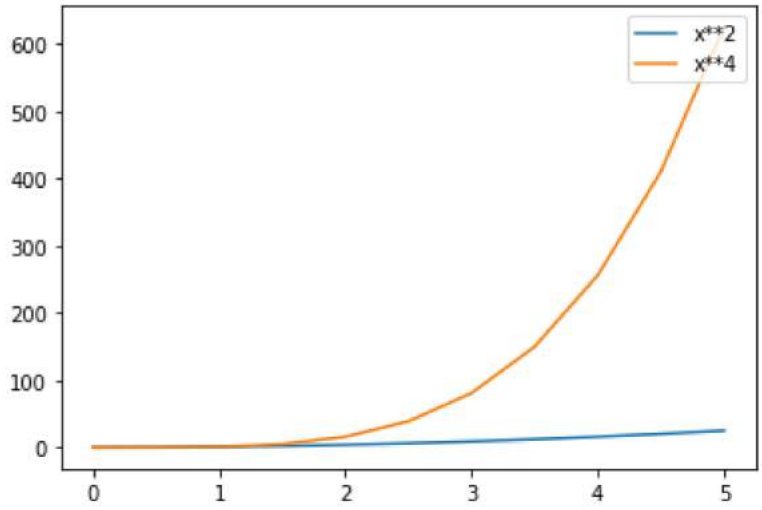

## ADDING LEGENDS TO GRAPH

```
f=p.figure()
axes=f.add_axes([0.5,0.5,0.8,0.8])
axes.plot(x,x**2,label="x**2")
axes.plot(x,x**4,label="x**4")
axes.legend()
```

```
<matplotlib.legend.Legend at 0x7f6adaa68f90>
```


```
f=p.figure()
axes=f.add_axes([0.5,0.5,0.8,0.8])
axes.plot(x,x**2,label="x**2")
axes.plot(x,x**4,label="x**4")
axes.legend(loc=1)
```

```
<matplotlib.legend.Legend at 0x7f6ad9bb7690>
```

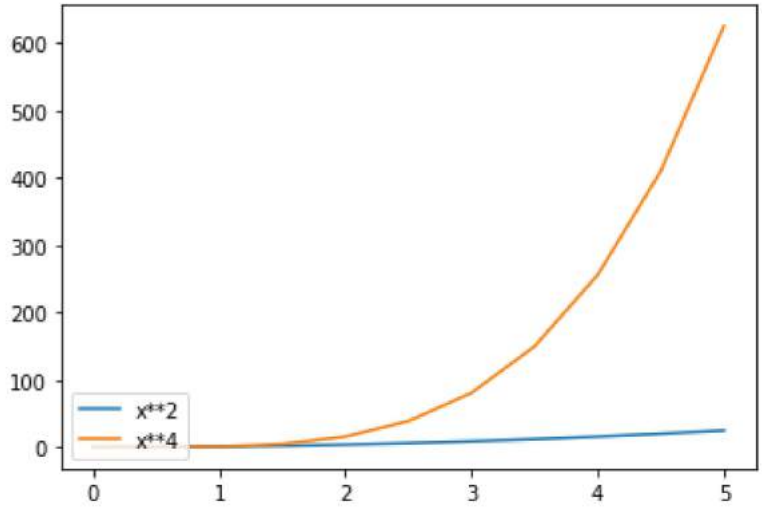



```
f=p.figure()
axes=f.add_axes([0.5,0.5,0.8,0.8])
axes.plot(x,x**2,label="x**2")
axes.plot(x,x**4,label="x**4")
axes.legend(loc=2)
```

```
<matplotlib.legend.Legend at 0x7f6ad9be2cd0>
```

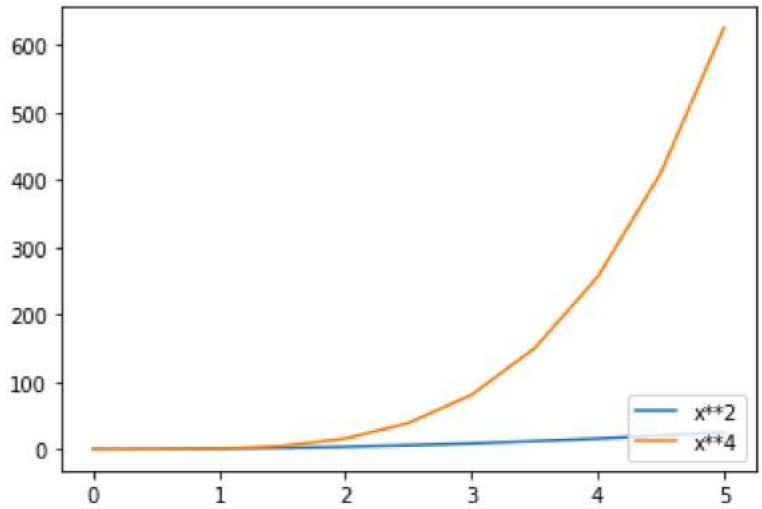

```
f=p.figure()
axes=f.add_axes([0.5,0.5,0.8,0.8])
axes.plot(x,x**2,label="x**2")
axes.plot(x,x**4,label="x**4")
axes.legend(loc=3)
```

```
<matplotlib.legend.Legend at 0x7f6ad98940d0>
```




```
f=p.figure()
axes=f.add_axes([0.5,0.5,0.8,0.8])
axes.plot(x,x**2,label="x**2")
axes.plot(x,x**4,label="x**4")
axes.legend(loc=4)
```

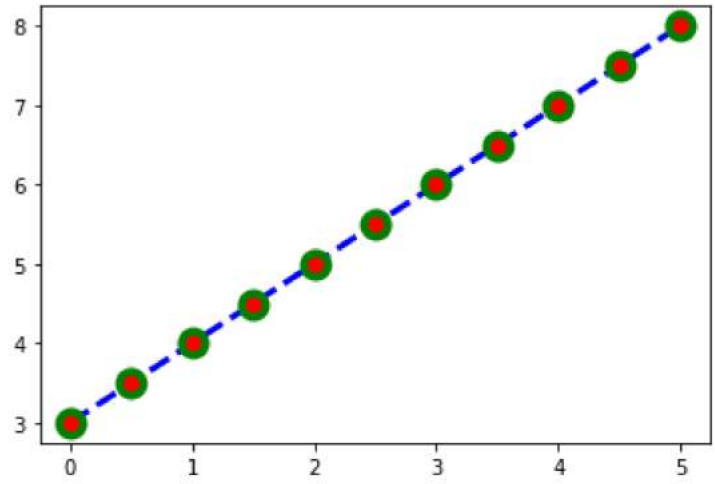
```
<matplotlib.legend.Legend at 0x7f6ad990fa50>
```




## LINECOLOR, LINEWIDTH, LINETYPE

```
p.plot(x,x+3,color='b',lw=3,ls='-',marker='o',markersize=12,linestyle='dashed',markeredgecolo
```

```
/usr/local/lib/python3.7/dist-packages/ipykernel_launcher.py:1: MatplotlibDeprecationWar
  """Entry point for launching an IPython kernel.
[<matplotlib.lines.Line2D at 0x7f5e0e51d5d0>]
```



```
import seaborn as s
```

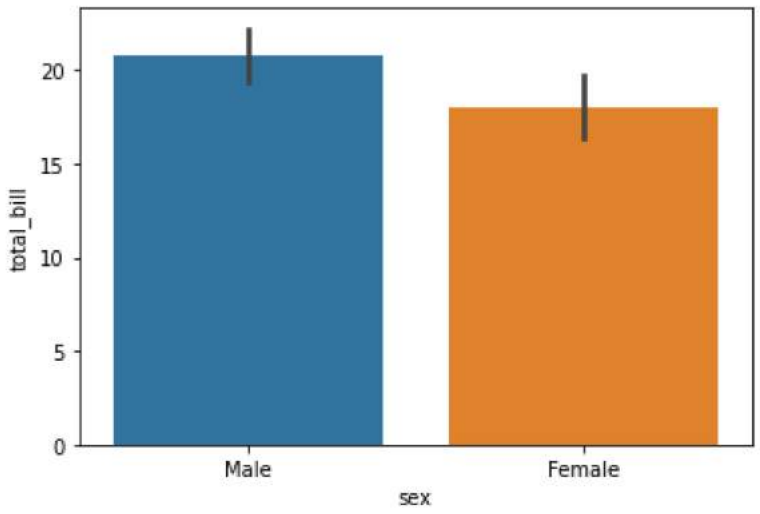
```
dataset=s.load_dataset('tips')
dataset.head()
```

| | 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 |

## BARPLOT

```
s.barplot(x='sex',y='total_bill',data=dataset)
```

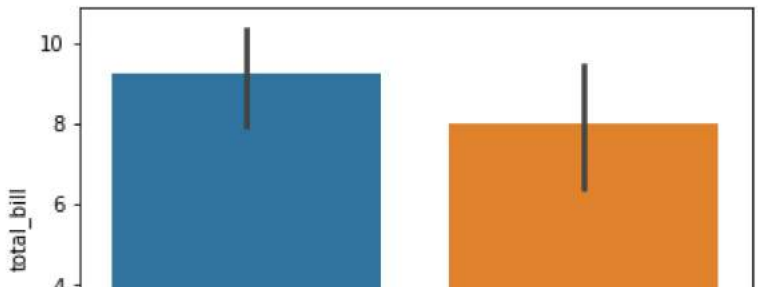
```
<matplotlib.axes._subplots.AxesSubplot at 0x7fd87322a850>
```




```
import numpy as n
```

```
s.barplot(x='sex',y='total_bill',data=dataset,estimator=n.std)
```

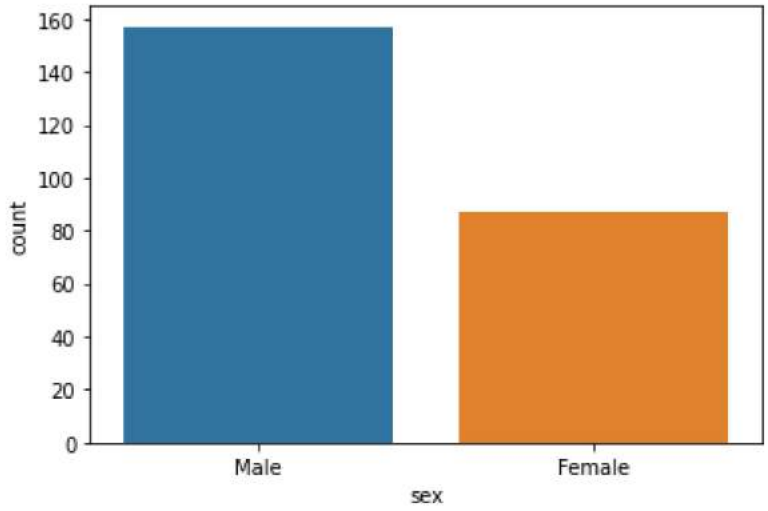
```
<matplotlib.axes._subplots.AxesSubplot at 0x7fd85e51ef10>
```




COUNTPLOT here y is implicitly set as count of x co ordinate

```
s.countplot(x='sex',data=dataset)
```

```
<matplotlib.axes._subplots.AxesSubplot at 0x7fd85e483050>
```




BOXPLOT

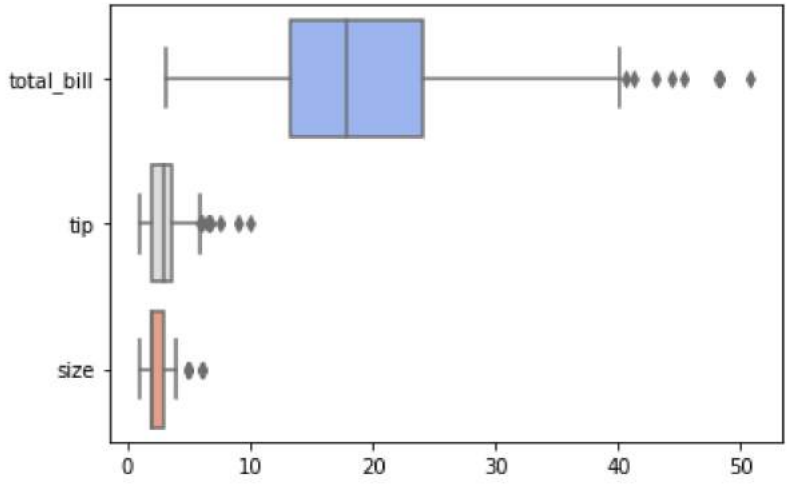
```
s.boxplot(x='day',y='total_bill',data=dataset,palette='rainbow')
```

```
<matplotlib.axes._subplots.AxesSubplot at 0x7fd85e3d9c10>
```

For horizontal representation

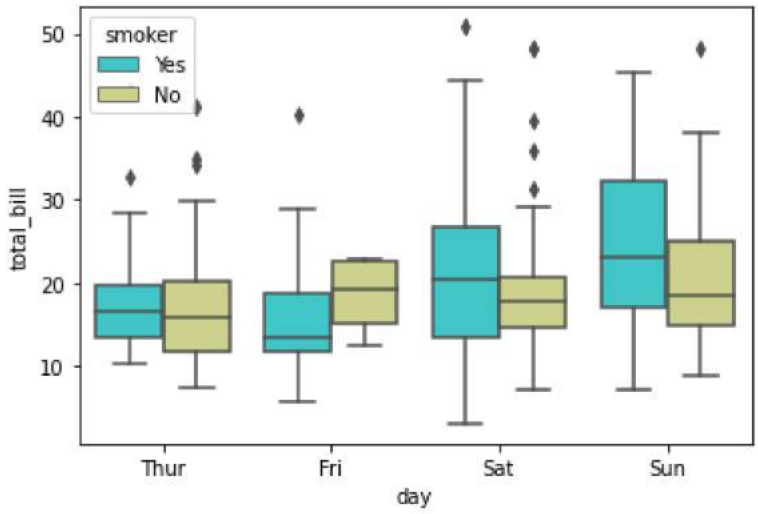
```
s.boxplot(data=dataset,palette='coolwarm',orient='h')
```

```
<matplotlib.axes._subplots.AxesSubplot at 0x7fd850d4ee10>
```




```
s.boxplot(x='day',y='total_bill',data=dataset,palette='rainbow',hue='smoker')
```

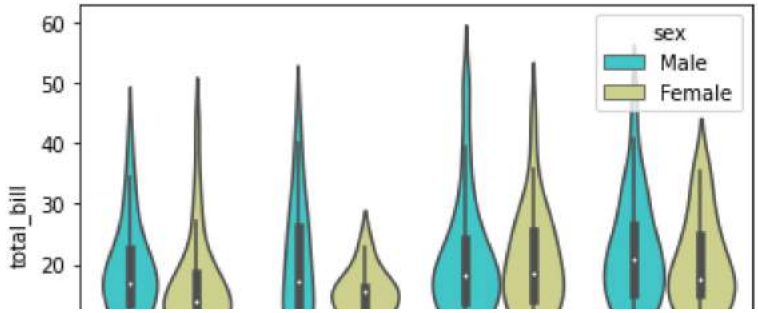
```
<matplotlib.axes._subplots.AxesSubplot at 0x7fd85cbf6410>
```




VIOLINPLOT

```
s.violinplot(x='day',y='total_bill',data=dataset,palette='rainbow',hue='sex')
```

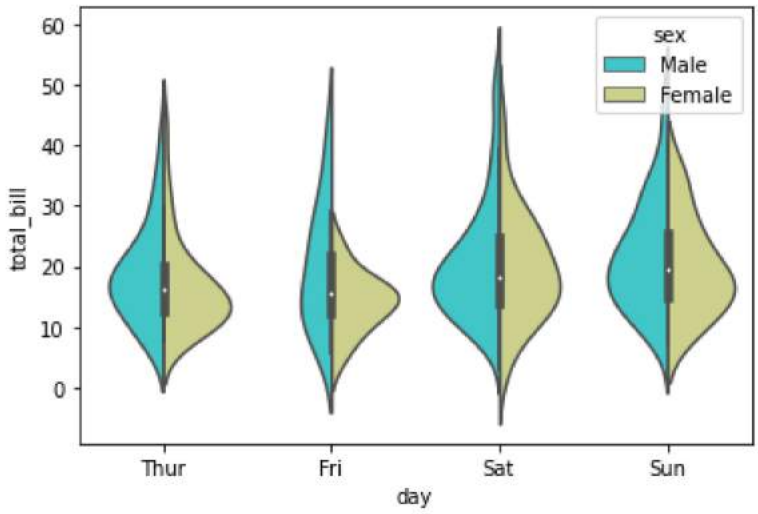
```
<matplotlib.axes._subplots.AxesSubplot at 0x7fd85cb747d0>
```




FOR COMPARISON

```
s.violinplot(x='day',y='total_bill',data=dataset,palette='rainbow',hue='sex',split=True)
```

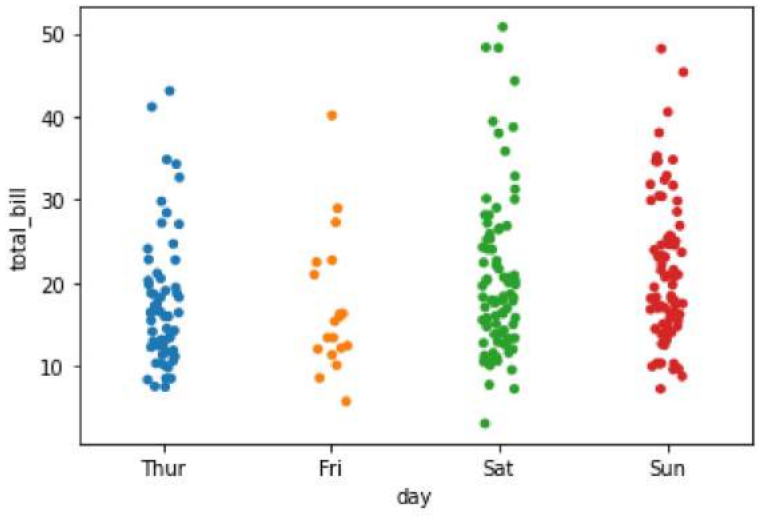
```
<matplotlib.axes._subplots.AxesSubplot at 0x7fd8597ccd10>
```




STRIP PLOT

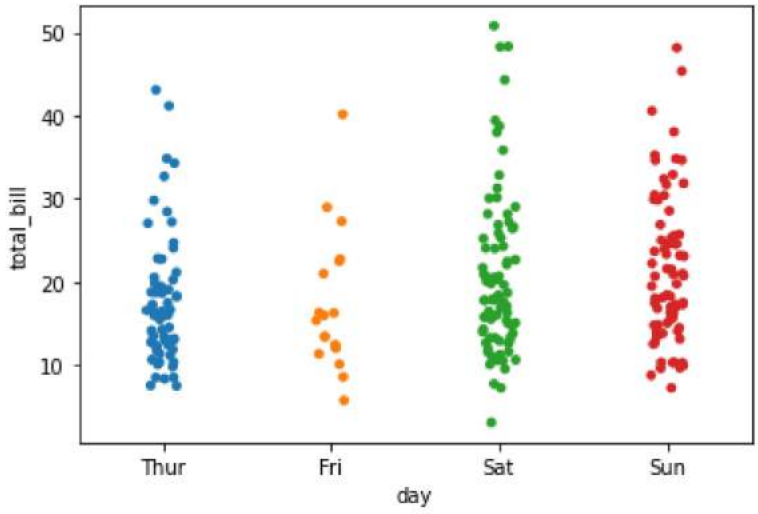
```
s.stripplot(x='day',y='total_bill',data=dataset)
```

```
<matplotlib.axes._subplots.AxesSubplot at 0x7fd859777350>
```



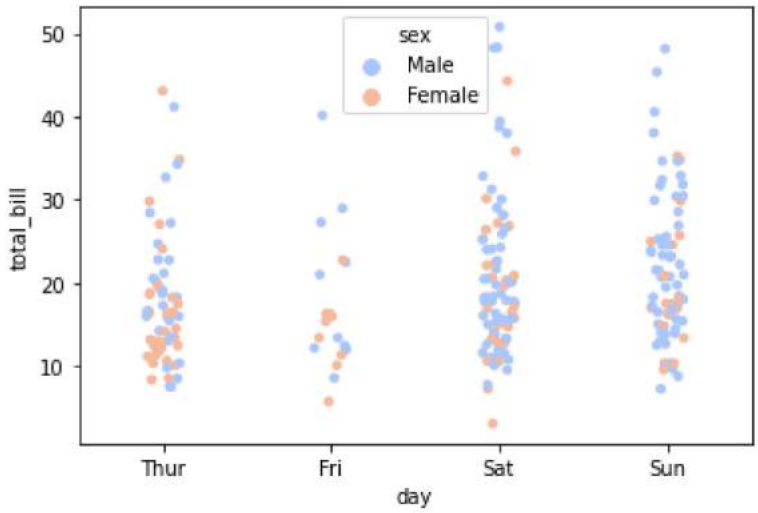
```
s.stripplot(x='day',y='total_bill',data=dataset,jitter=True)
```

```
<matplotlib.axes._subplots.AxesSubplot at 0x7fd85971f510>
```




```
s.stripplot(x='day',y='total_bill',data=dataset,jitter=True,hue='sex',palette='coolwarm')
```

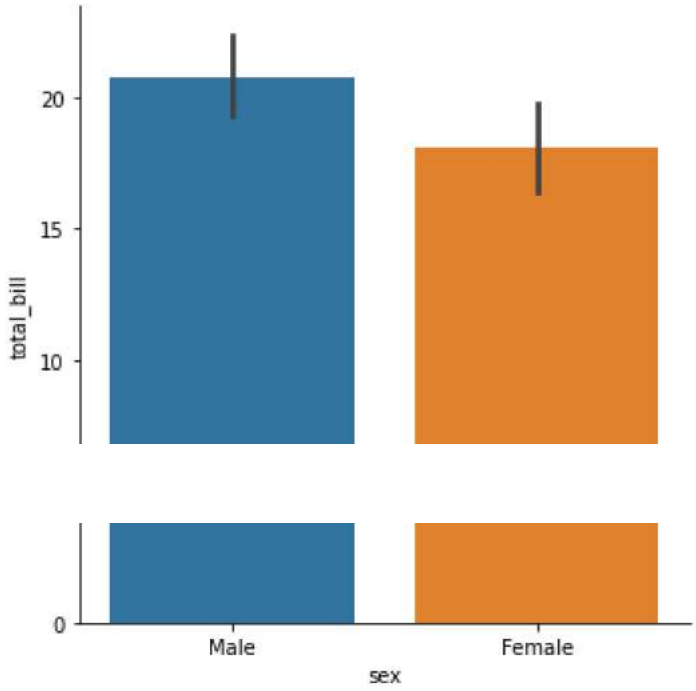
```
<matplotlib.axes._subplots.AxesSubplot at 0x7fd85971c750>
```




## FACTORPLOT

```
s.factorplot(x='sex',y='total_bill',data=dataset,kind='bar')
```

```
/usr/local/lib/python3.7/dist-packages/seaborn/categorical.py:3717: UserWarning: The `fa
  warnings.warn(msg)
<seaborn.axisgrid.FacetGrid at 0x7fd859771ad0>
```



## DISTPLOT

```
import seaborn as s
```

## LOADING DATA SET CALLED 'tips'

```
dataset=s.load_dataset('tips')
```

```
dataset.head(10)
```

| | 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 |
| **5** | 25.29 | 4.71 | Male | No | Sun | Dinner | 4 |
| **6** | 8.77 | 2.00 | Male | No | Sun | Dinner | 2 |
| **7** | 26.88 | 3.12 | Male | No | Sun | Dinner | 4 |
| **8** | 15.04 | 1.96 | Male | No | Sun | Dinner | 2 |
| **9** | 14.78 | 3.23 | Male | No | Sun | Dinner | 2 |

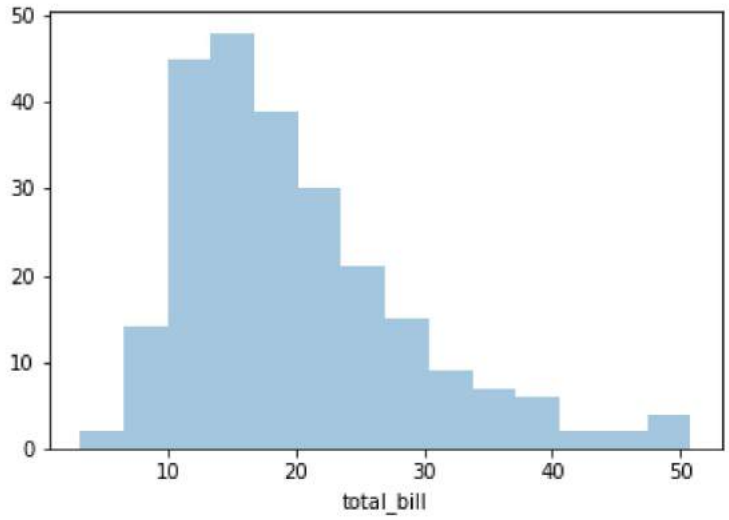
## Uinsg distplot(univariate)

```
s.distplot(dataset['tip'])
```

```
/usr/local/lib/python3.7/dist-packages/seaborn/distributions.py:2619: FutureWarning: `di
  warnings.warn(msg, FutureWarning)
<matplotlib.axes._subplots.AxesSubplot at 0x7fcae50a3950>
```

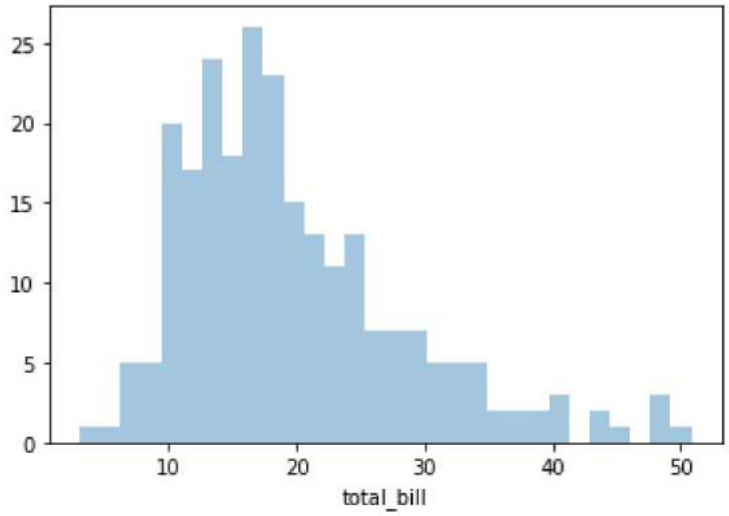

```
s.distplot(dataset['total_bill'],kde=False)
```

```
/usr/local/lib/python3.7/dist-packages/seaborn/distributions.py:2619: FutureWarning: `di
  warnings.warn(msg, FutureWarning)
<matplotlib.axes._subplots.AxesSubplot at 0x7fcad67ce890>
```



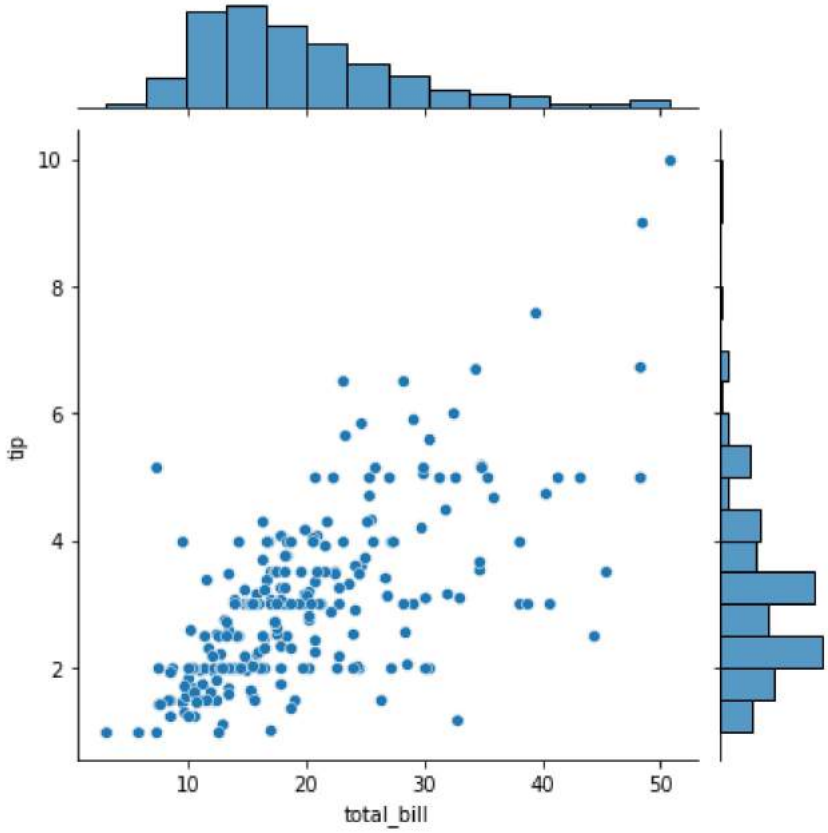

```
s.distplot(dataset['total_bill'],kde=False,bins=30)
```

```
/usr/local/lib/python3.7/dist-packages/seaborn/distributions.py:2619: FutureWarning: `di
  warnings.warn(msg, FutureWarning)
<matplotlib.axes._subplots.AxesSubplot at 0x7fcad67c15d0>
```



```
s.jointplot(x='total_bill',y='tip',data=dataset,kind='scatter')
```

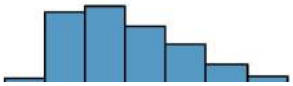
```
<seaborn.axisgrid.JointGrid at 0x7fcae1cdd790>
```




hex=density

```
s.jointplot(x='total_bill',y='tip',data=dataset,kind='hex')
```

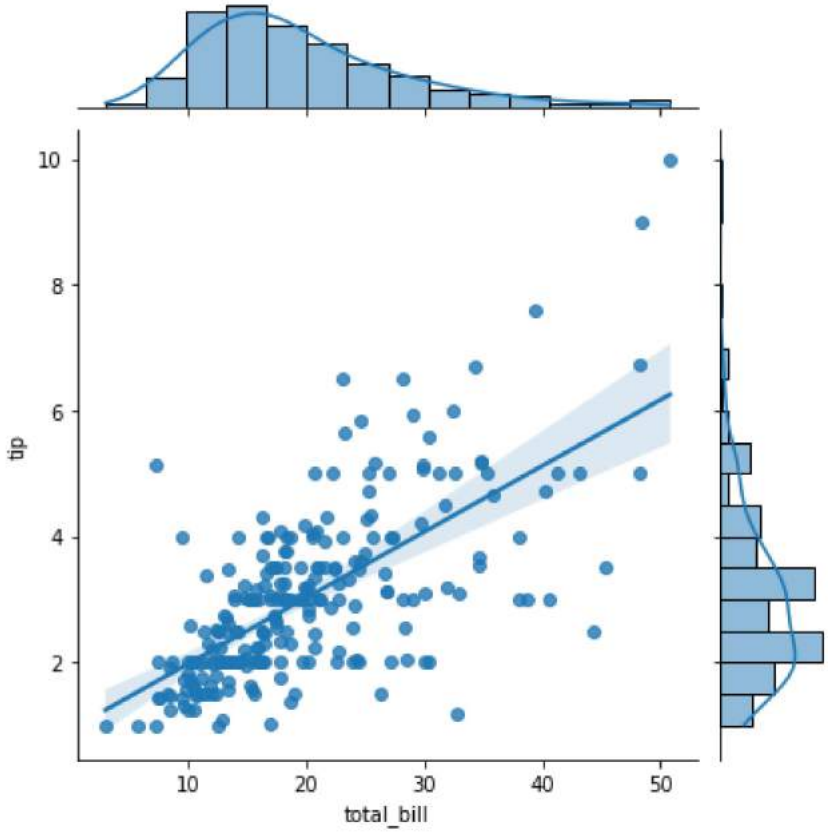
```
<seaborn.axisgrid.JointGrid at 0x7fcaea125390>
```



reg=regression

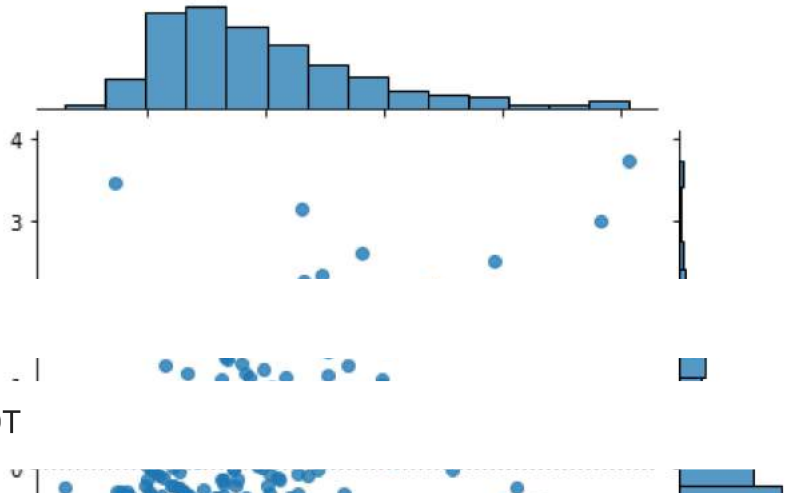
```
s.jointplot(x='total_bill',y='tip',data=dataset,kind='reg')
```

```
<seaborn.axisgrid.JointGrid at 0x7fcad8d2ced0>
```




```
s.jointplot(x='total_bill',y='tip',data=dataset,kind='resid')
```

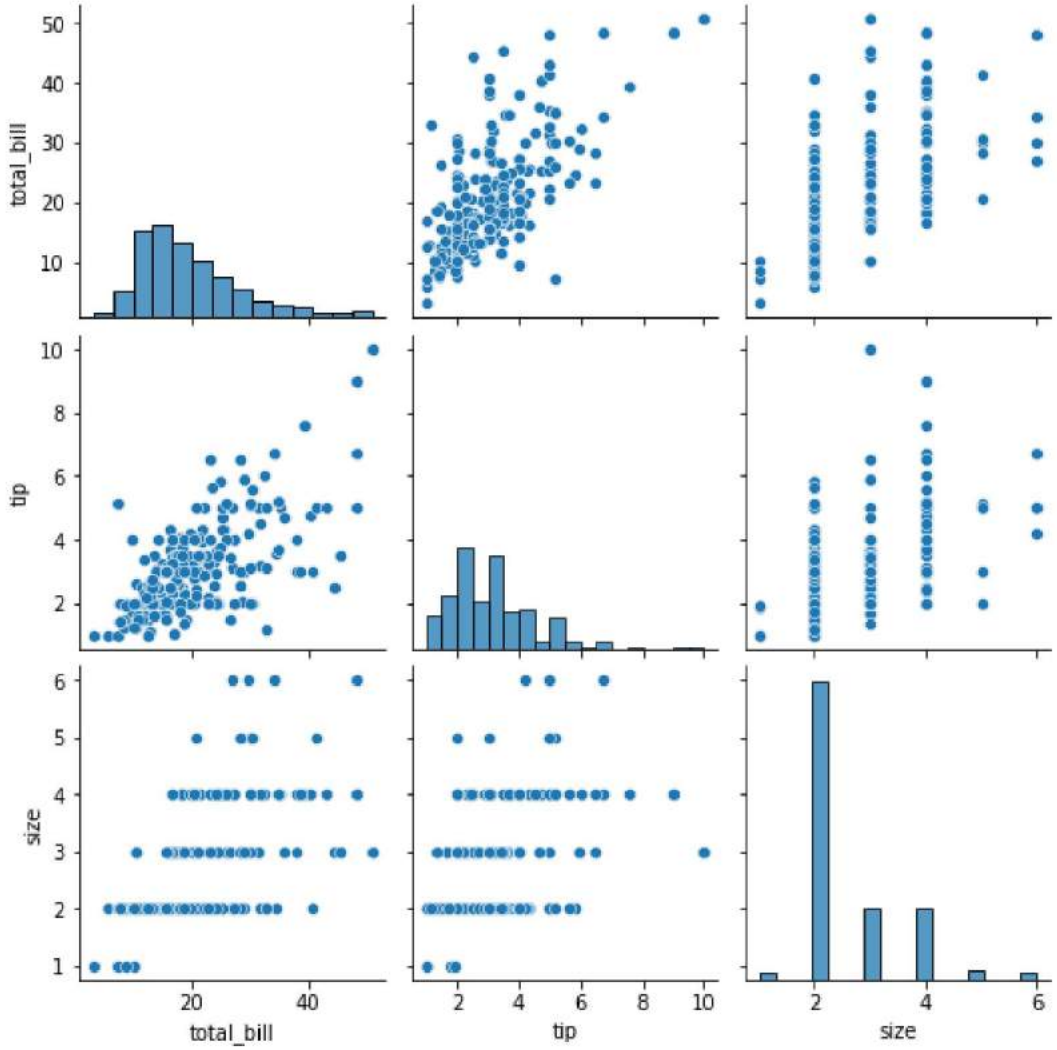
```
<seaborn.axisgrid.JointGrid at 0x7fcad8c1f3d0>
```



PAIRPLOT

```
s.pairplot(dataset)
```

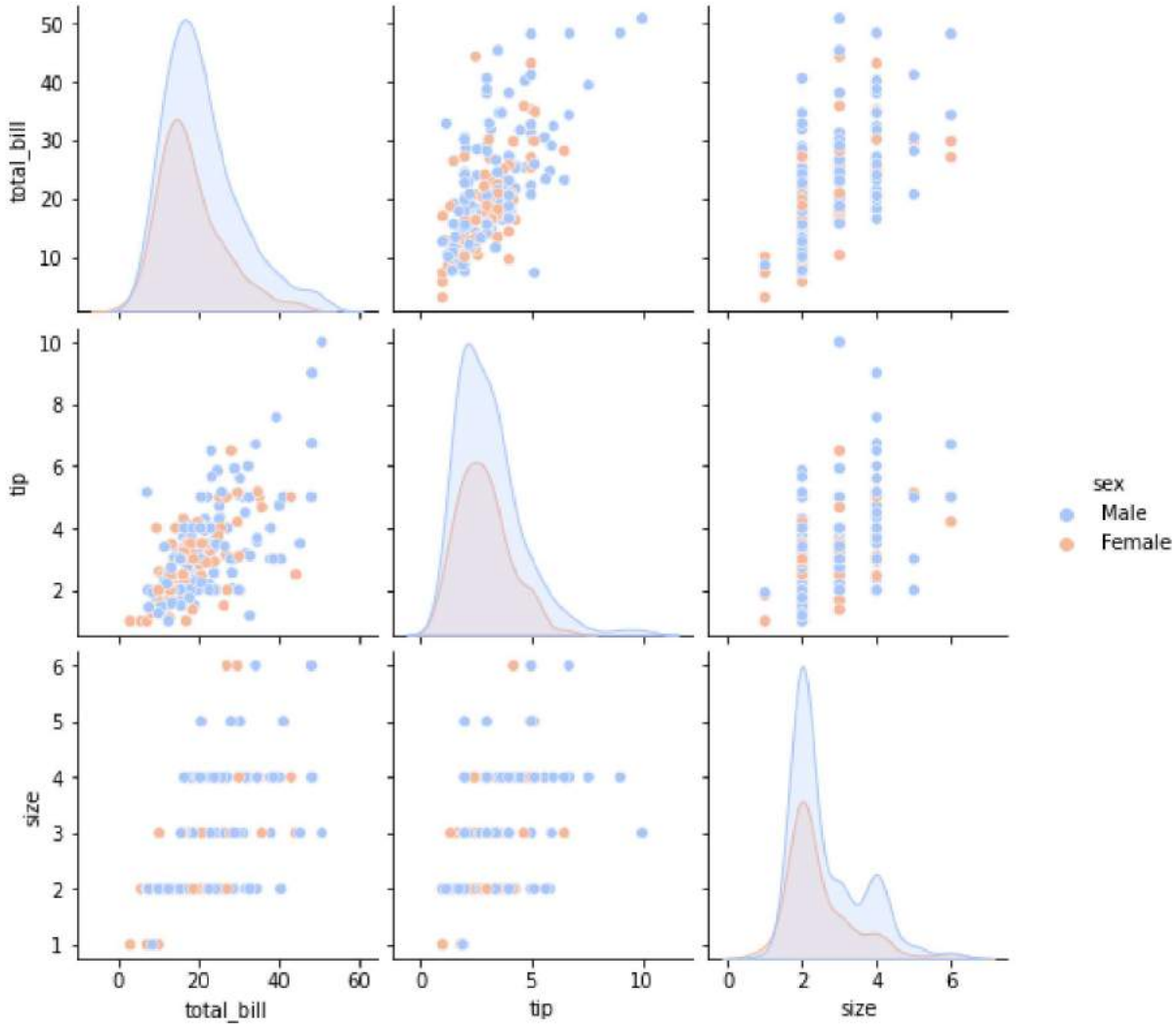
```
<seaborn.axisgrid.PairGrid at 0x7fcad8a70890>
```




To identify the geneder type for each data point

```
s.pairplot(dataset,hue='sex',palette='coolwarm')
```

```
<seaborn.axisgrid.PairGrid at 0x7fcad85db710>
```

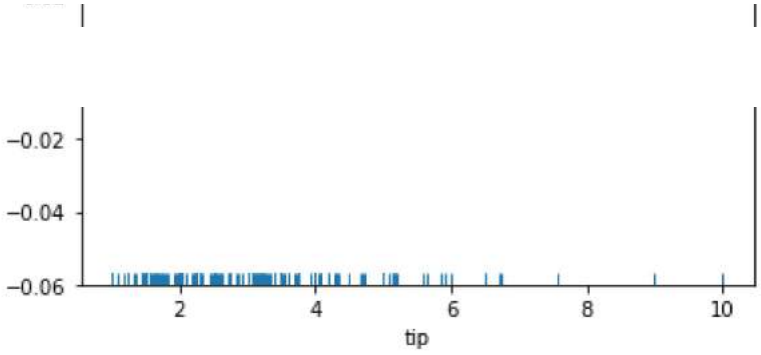



## RUGPLOT

```
s.rugplot(dataset['tip'])
```

```
<matplotlib.axes._subplots.AxesSubplot at 0x7fcad8a26050>
```


Also univariant, only creates a dashed line, KDE is implememented using RUGPLOT Very important



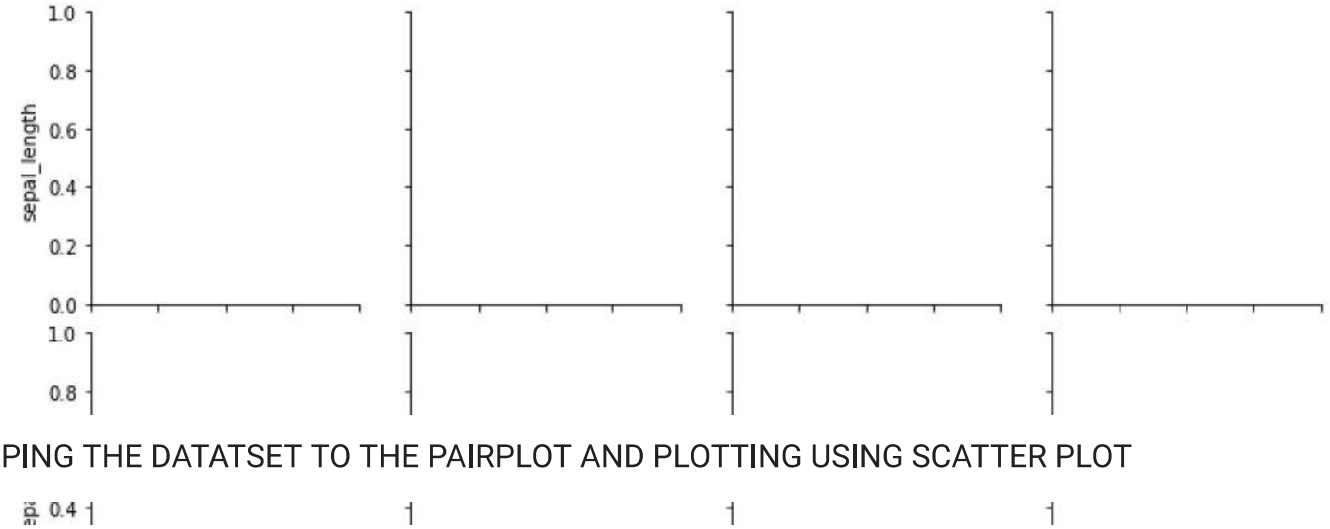
```
import seaborn as s
import matplotlib.pyplot as p


iris_dataset=s.load_dataset('iris')
iris_dataset.head(5)
```

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

PAIR PLOT

```
s.PairGrid(iris_dataset)
```

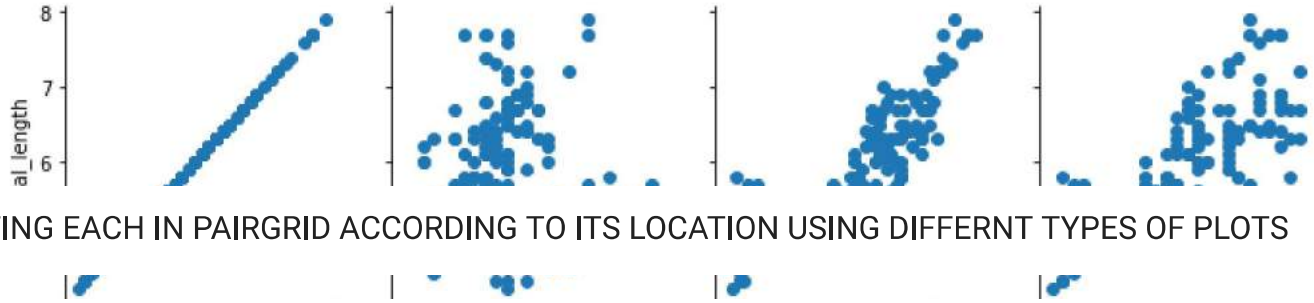
```
<seaborn.axisgrid.PairGrid at 0x7f51ad88a390>
```




MAPPING THE DATATSET TO THE PAIRPLOT AND PLOTTING USING SCATTER PLOT

```
mapping=s.PairGrid(iris_dataset)
mapping.map(p.scatter)
```

```
<seaborn.axisgrid.PairGrid at 0x7f51a3e02150>
```




PLOTTING EACH IN PAIRGRID ACCORDING TO ITS LOCATION USING DIFFERNT TYPES OF PLOTS



```
mapping=s.PairGrid(iris_dataset)
mapping.map_diag(p.hist)
mapping.map_upper(p.scatter)
mapping.map_lower(s.kdeplot)
```

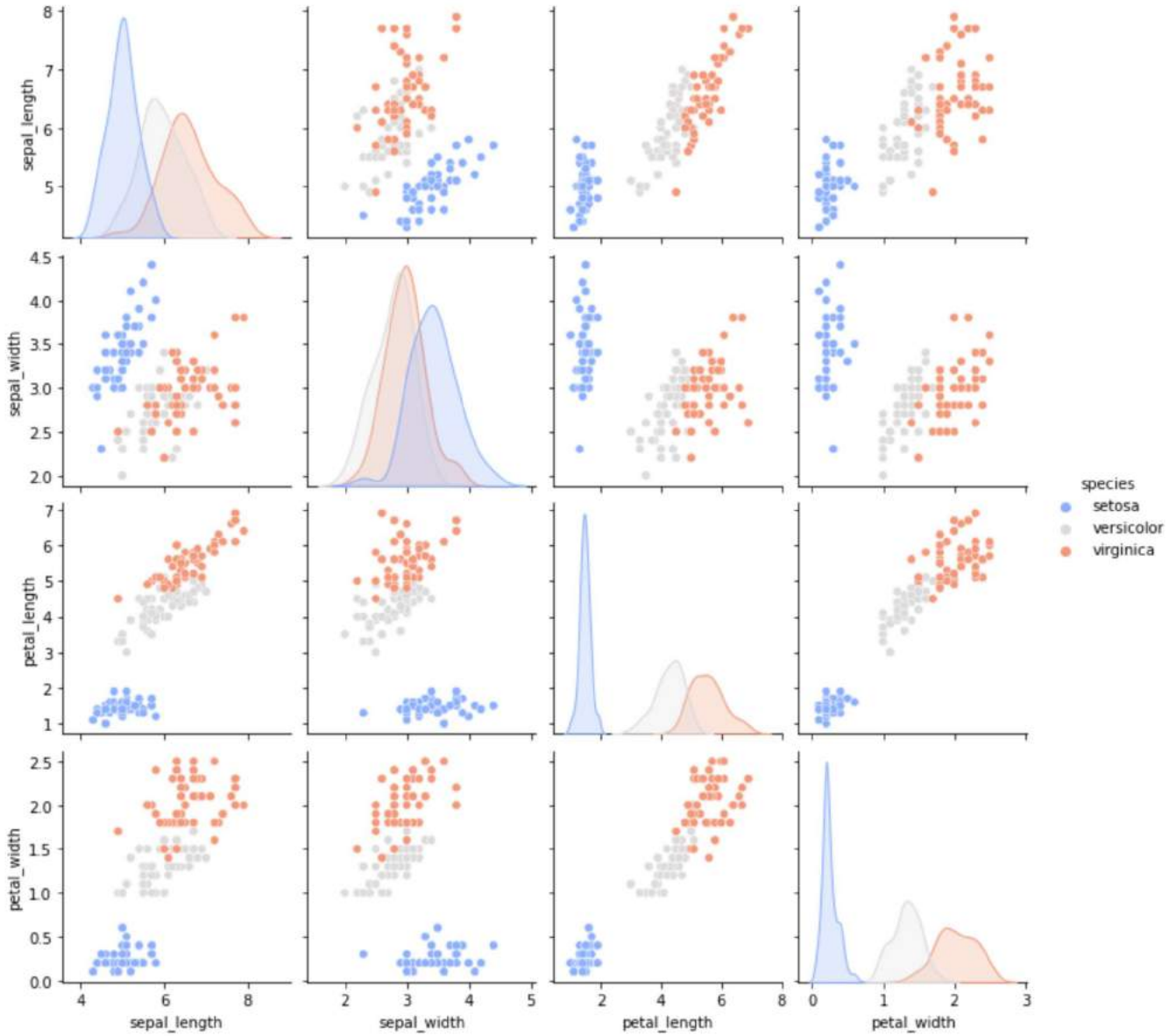
```
<seaborn.axisgrid.PairGrid at 0x7f51a198b3d0>
```

Using hue to seperate different types of species



```
s.pairplot(iris_dataset,hue='species',palette='coolwarm')
```

```
<seaborn.axisgrid.PairGrid at 0x7f51a18a5c90>
```




FACET GRID

```
tips_dataset=s.load_dataset('tips')
tips_dataset.head(5)
```

| | 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 |

MAPPING DATASET

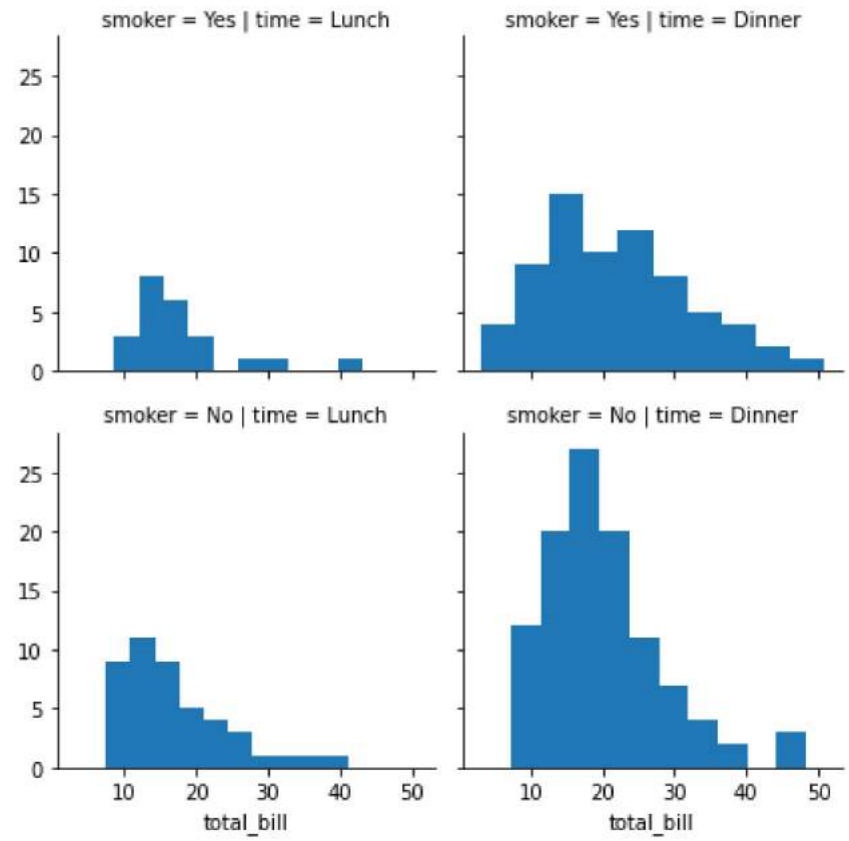
=>here the graph 1 represe the thotal_bill trend for smokers at lunch time

graph 2: total_bill for a smoker at dinner time

graph 3:total_bill for non smoker at lunch time<br<

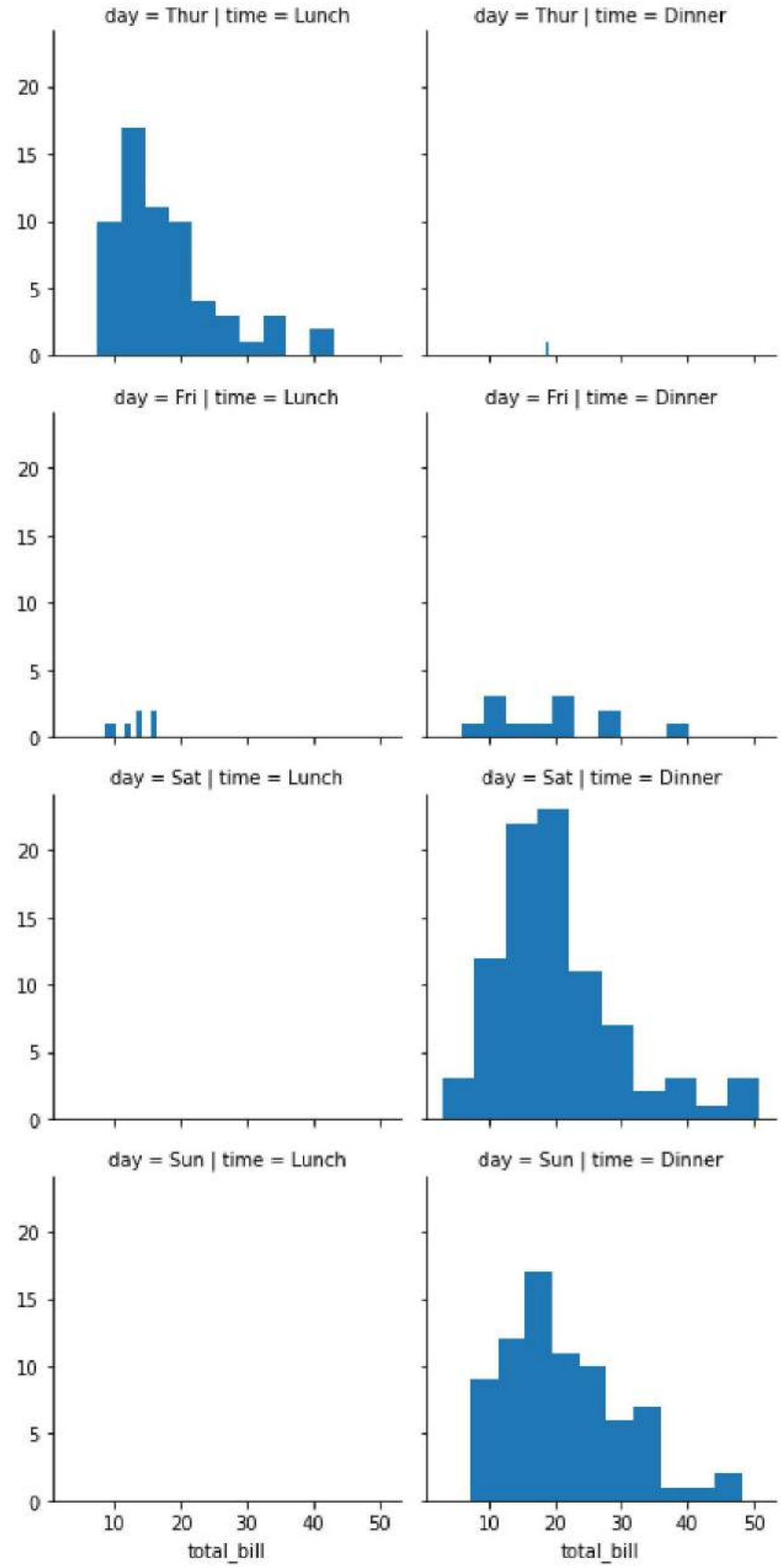
```
mapping=s.FacetGrid(tips_dataset,col='time',row='smoker')
mapping.map(p.hist,'total_bill')
```

```
<seaborn.axisgrid.FacetGrid at 0x7f519cfafd90>
```



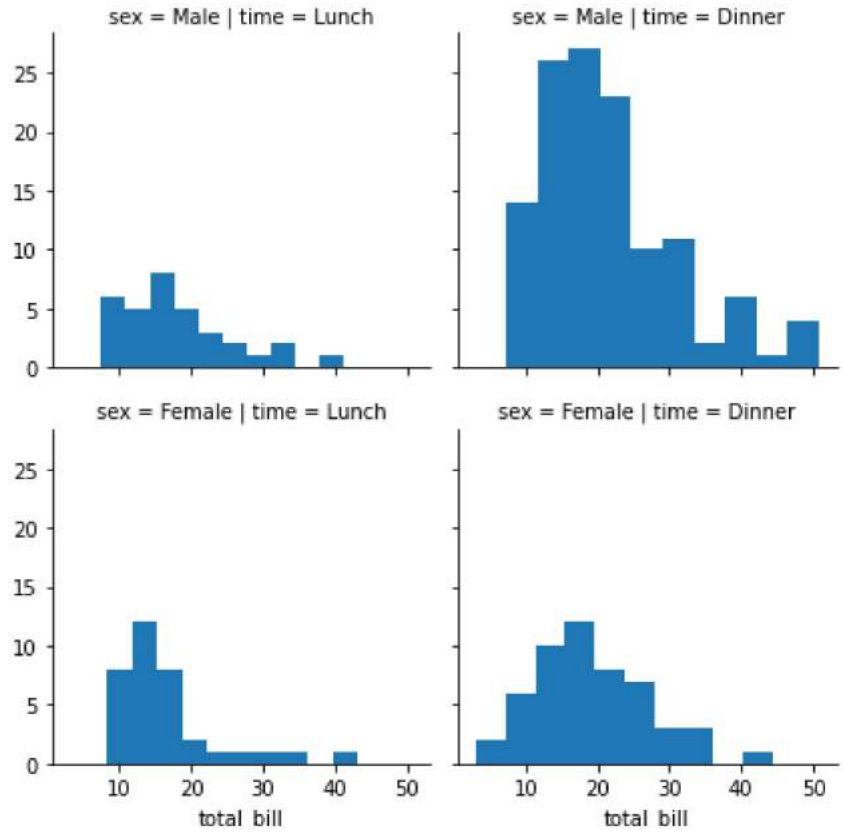

```
mapping=s.FacetGrid(tips_dataset,col='time',row='day')
mapping.map(p.hist,'total_bill')
```

```
<seaborn.axisgrid.FacetGrid at 0x7f519ce6b9d0>
```




```
mapping=s.FacetGrid(tips_dataset,col='time',row='sex')
mapping.map(p.hist,'total_bill')
```

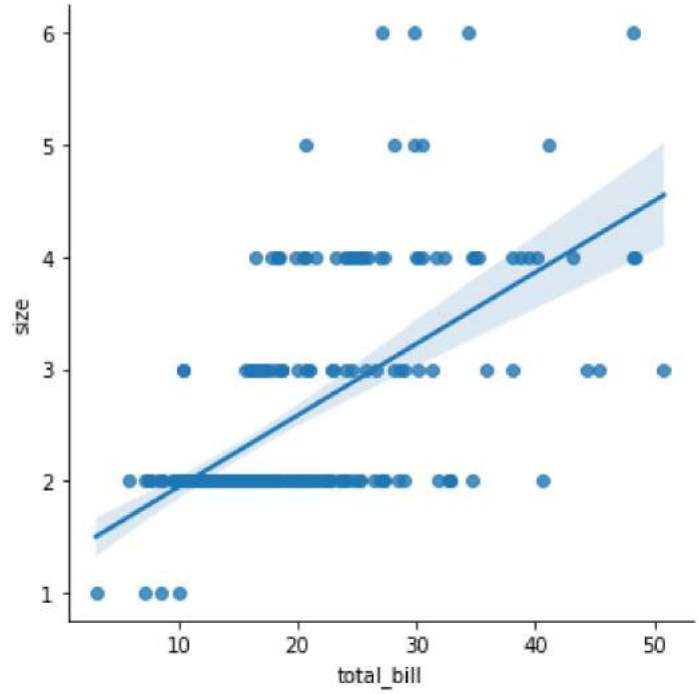
```
<seaborn.axisgrid.FacetGrid at 0x7f519d188110>
```




## REGRESSION PLOT

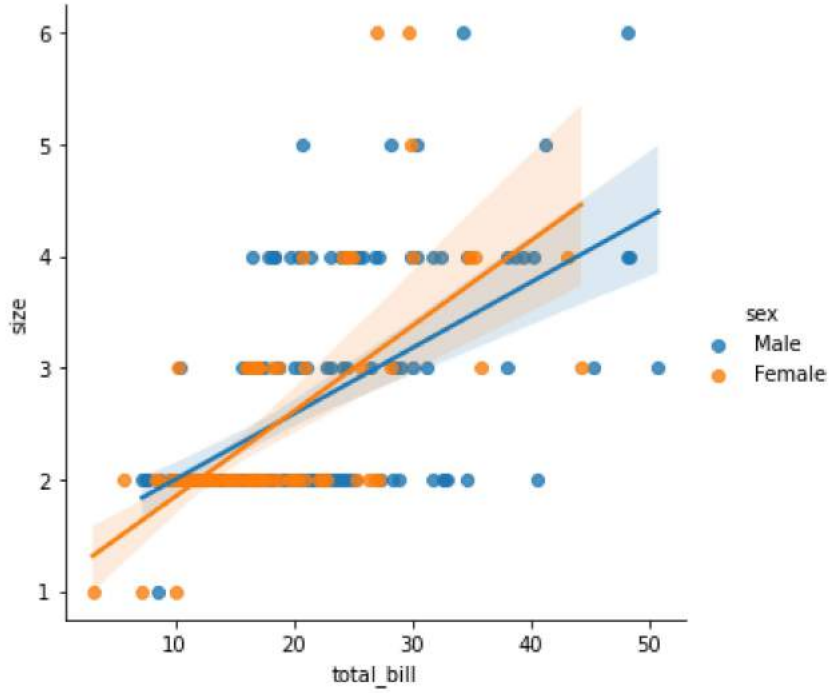
```
s.lmplot(x='total_bill',y='size' ,data=tips_dataset)
```

```
<seaborn.axisgrid.FacetGrid at 0x7f519ca04910>
```




```
s.lmplot(x='total_bill',y='size',hue='sex',data=tips_dataset)
```

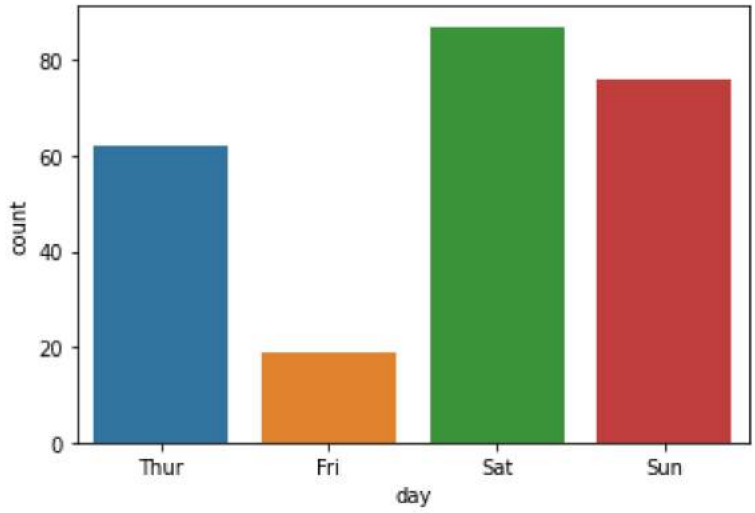
```
<seaborn.axisgrid.FacetGrid at 0x7f519c9a6b10>
```




## COUNT PLOT

```
s.countplot(x='day', data=tips_dataset)
```

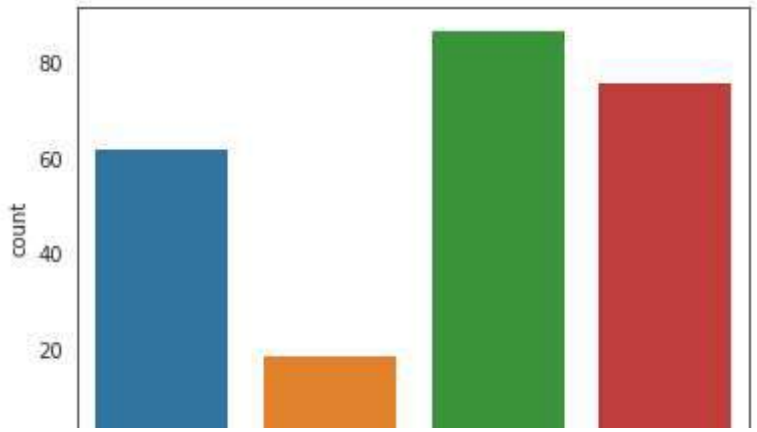
```
<matplotlib.axes._subplots.AxesSubplot at 0x7f519c8e5f50>
```




## TO MAKE BACKGROUND AS WHITE

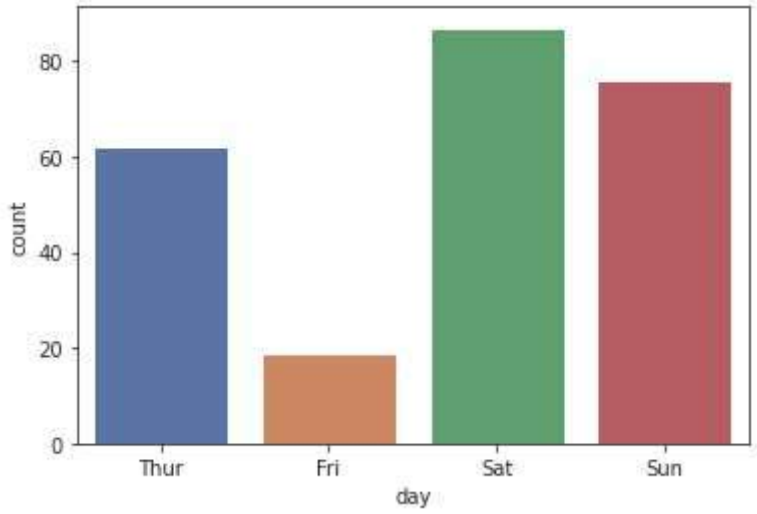
```
s.set_style('white')
s.countplot(x='day', data=tips_dataset)
```

```
<matplotlib.axes._subplots.AxesSubplot at 0x7f519c8aa910>
```

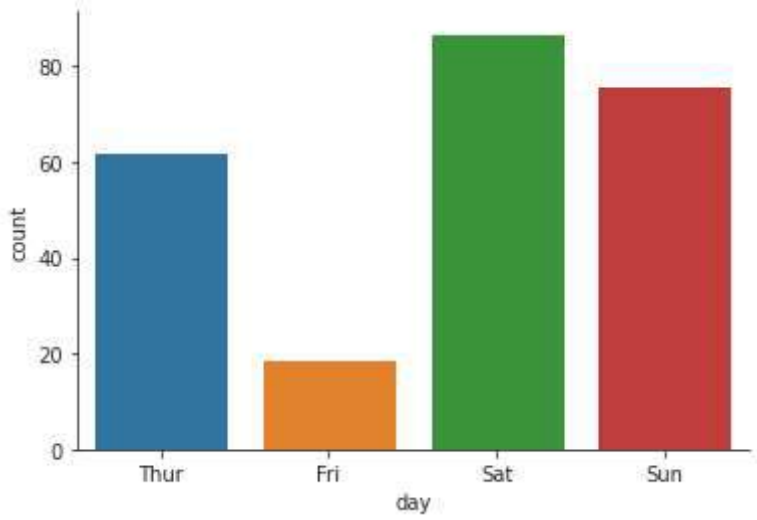



```
s.set_style('ticks')
s.countplot(x='day', data=tips_dataset,palette='deep')
```

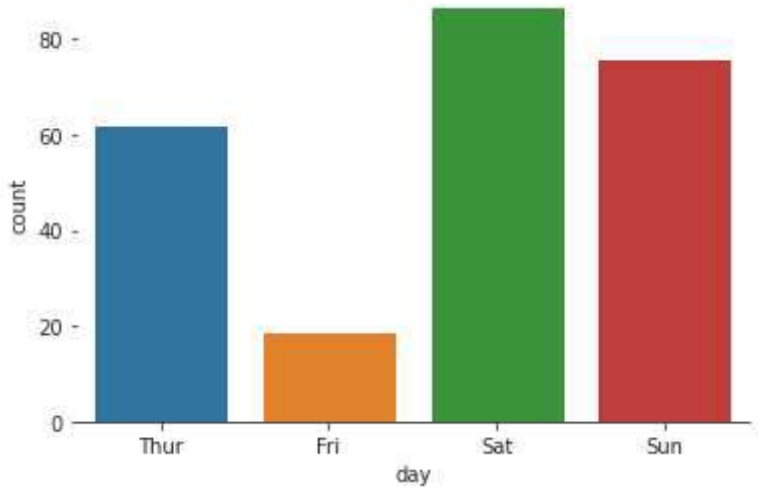
```
<matplotlib.axes._subplots.AxesSubplot at 0x7f519c86afd0>
```




```
s.countplot(x='day', data=tips_dataset)
s.despine()
```



```
s.countplot(x='day', data=tips_dataset)
s.despine(left='true')
```




```
s.countplot(x='day', data=tips_dataset)
s.despine(bottom='true',left='true')
```