

```

import matplotlib.pyplot as plt
import numpy as np
import seaborn as sns
%matplotlib inline

# Set Default size and color
plt.rcParams['font.size']=16
plt.rcParams['figure.figsize']=(10,5)
plt.rcParams['figure.facecolor']='white'
sns.set_style('whitegrid')

```

Line Plot

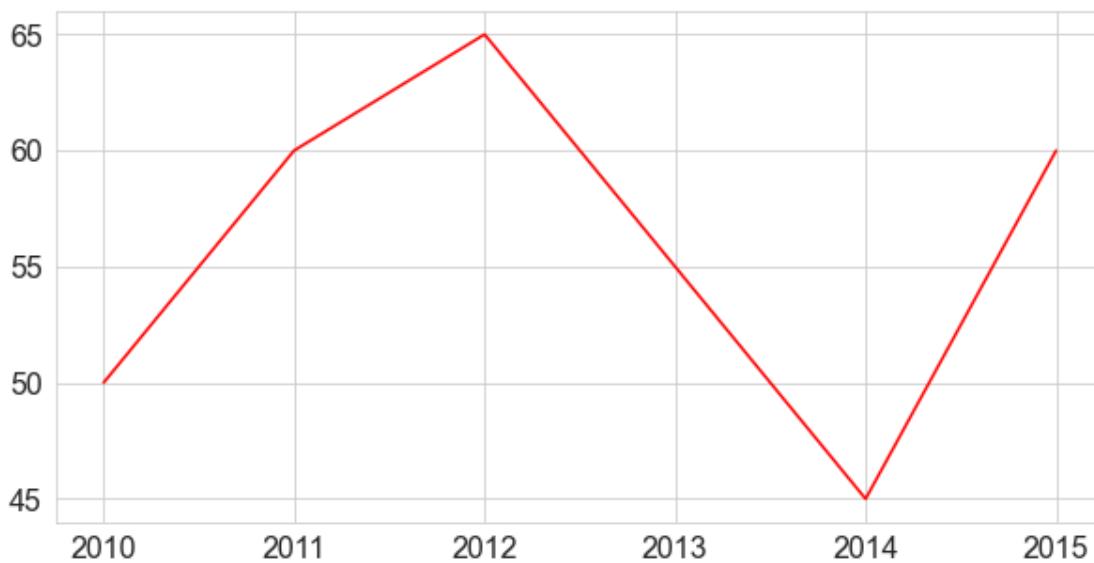
```

x = [2010,2011,2012,2013,2014,2015]
y = [50,60,65,55,45,60]

```

```
plt.plot(x,y,c='r')
```

```
[<matplotlib.lines.Line2D at 0x1f0d5c9a100>]
```

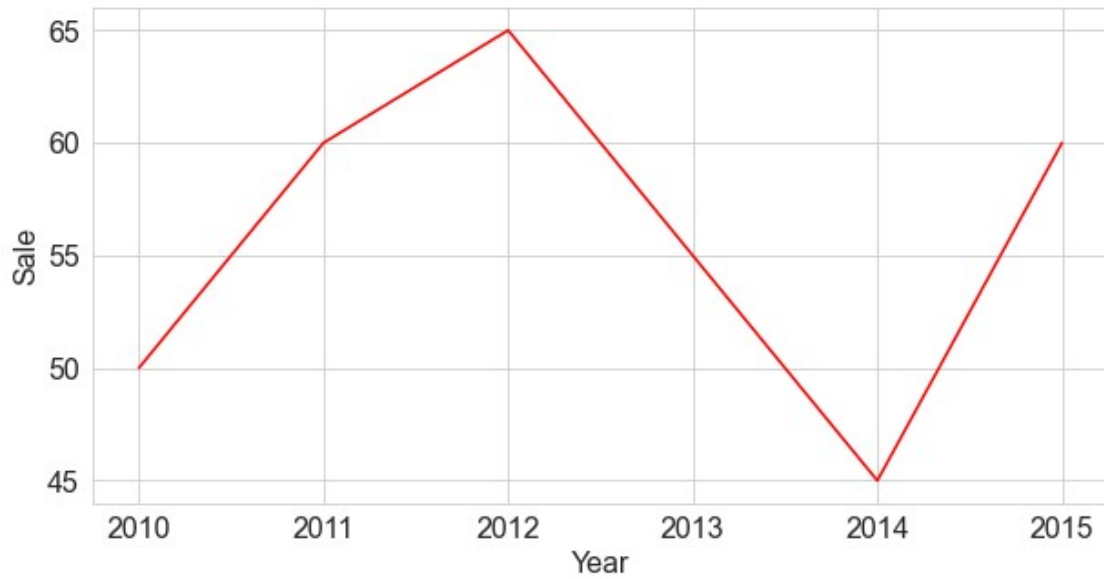


```

plt.plot(x,y,c='r');#remove 1st line
plt.xlabel("Year")
plt.ylabel("Sale")

```

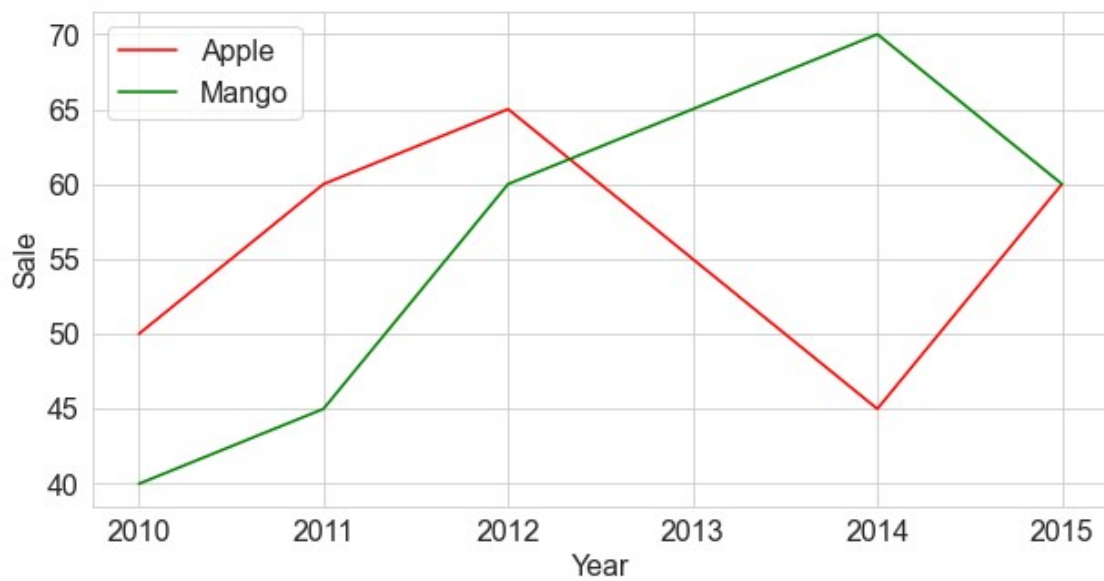
```
Text(0, 0.5, 'Sale')
```



```
y1 = [40,45,60,65,70,60]
```

```
plt.plot(x,y,c='r');#remove 1st line  
plt.plot(x,y1,c='g');  
plt.xlabel("Year")  
plt.ylabel("Sale")  
plt.legend(['Apple','Mango'])
```

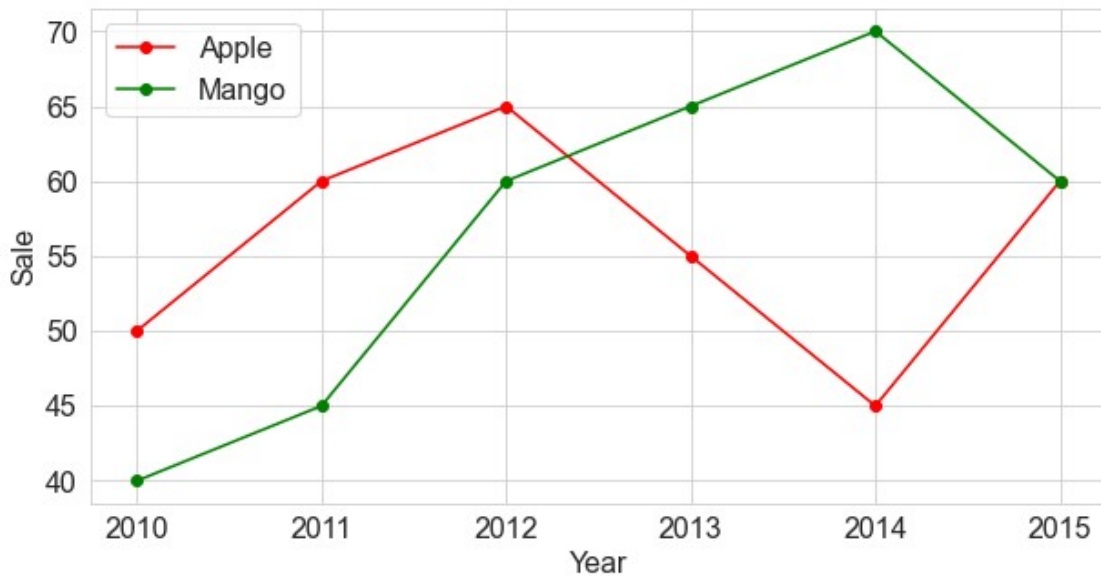
<matplotlib.legend.Legend at 0x1f0d8da3c70>



Add market in line plot

```
plt.plot(x,y,c='r',marker="o");#remove 1st line  
plt.plot(x,y1,c='g',marker="o");  
plt.xlabel("Year")  
plt.ylabel("Sale")  
plt.legend(['Apple','Mango'])
```

<matplotlib.legend.Legend at 0x1f0d8e09550>

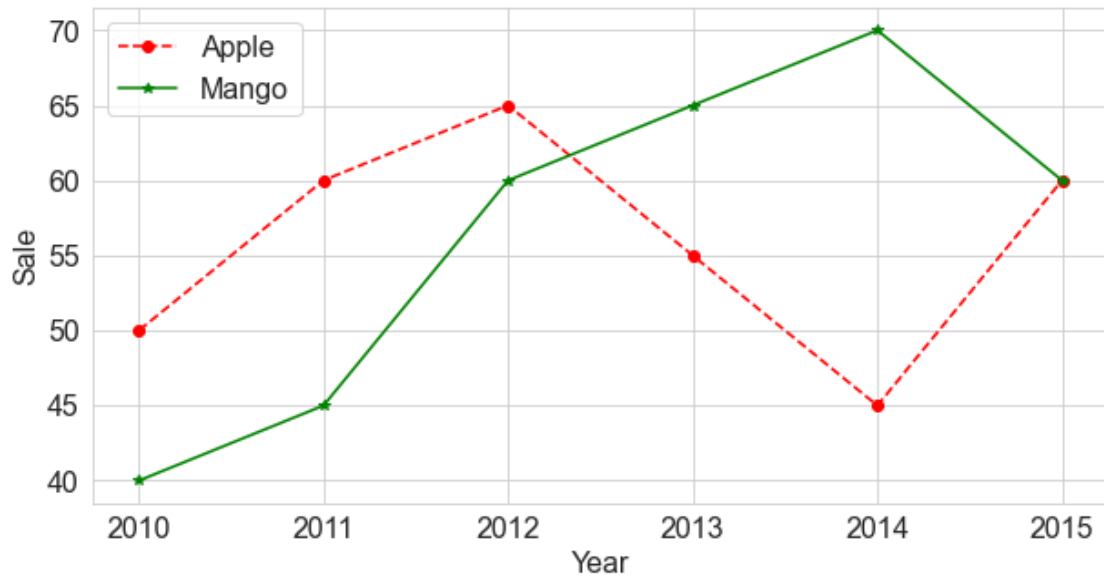


fmt shortform [marker,line,color]

'o--r' = marker='o',line='--',color='r'

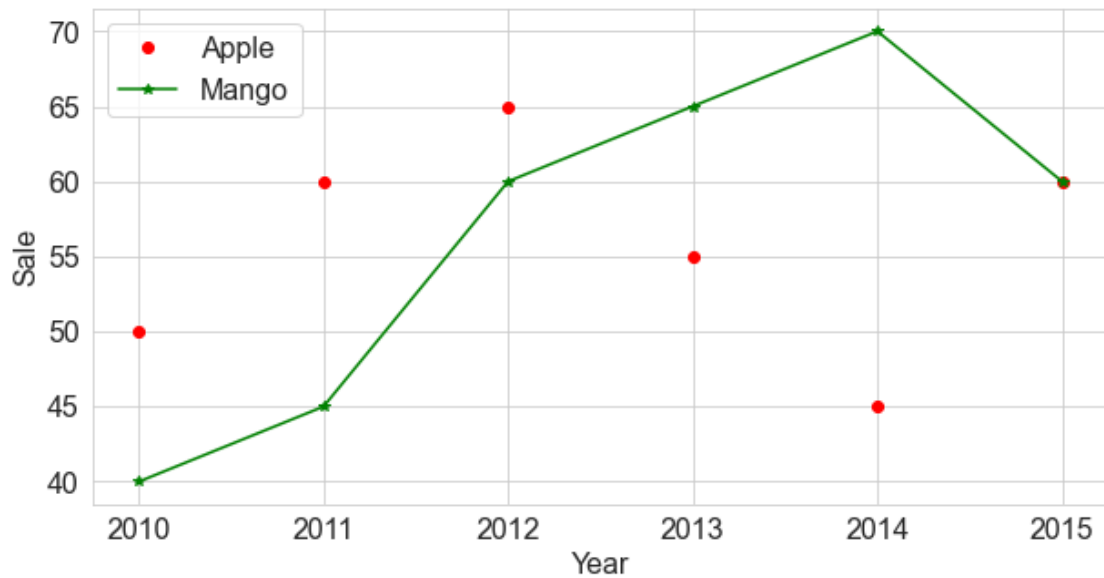
```
plt.plot(x,y,'o--r');#remove 1st line  
plt.plot(x,y1,'*-g');  
plt.xlabel("Year")  
plt.ylabel("Sale")  
plt.legend(['Apple','Mango'])
```

<matplotlib.legend.Legend at 0x1f0d8e70c40>



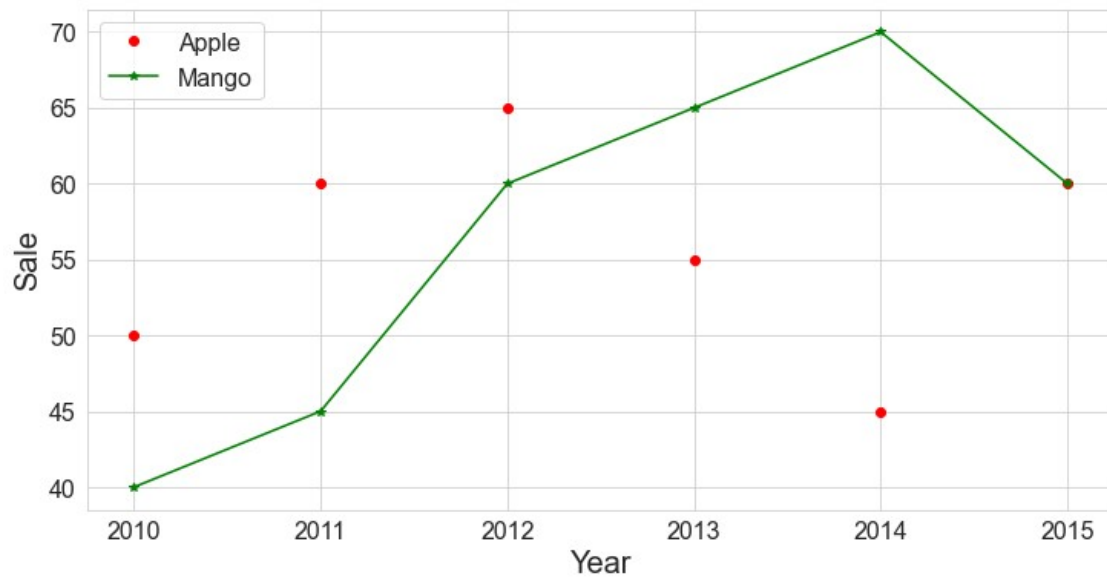
```
plt.plot(x,y,'or');#remove 1st line
plt.plot(x,y1,'*-g');
plt.xlabel("Year")
plt.ylabel("Sale")
plt.legend(['Apple', 'Mango'])
```

<matplotlib.legend.Legend at 0x1f0d8edb5b0>



```
plt.figure(figsize=(12,6))
plt.plot(x,y,'or');#remove 1st line
plt.plot(x,y1,'*-g');
plt.xlabel("Year",size='20')
plt.ylabel("Sale",size='20')
plt.legend(['Apple', 'Mango'])
```

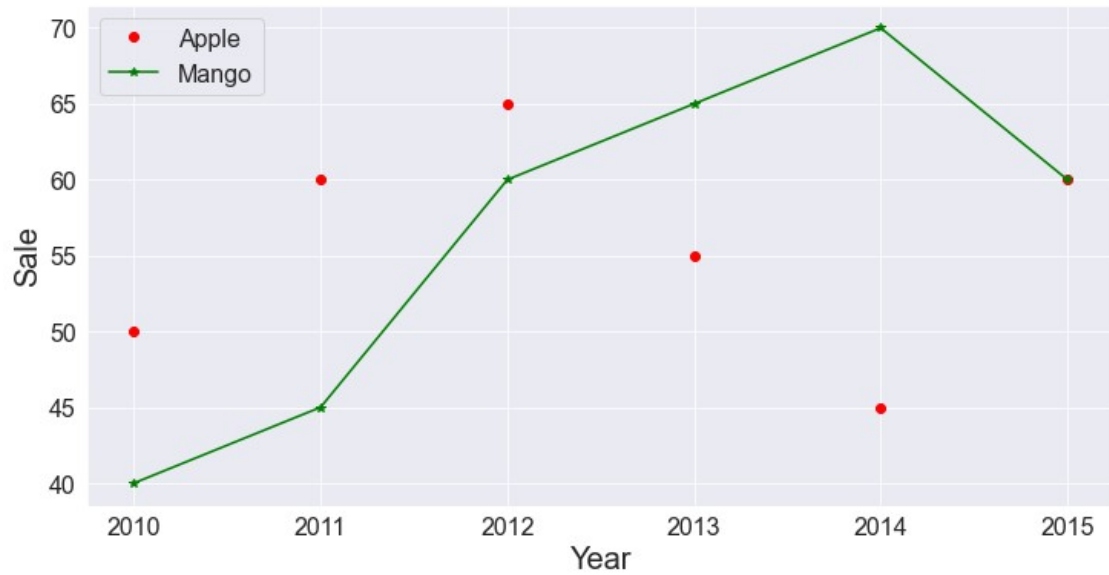
<matplotlib.legend.Legend at 0x1f0d8f328e0>



Set Grid using seaborn

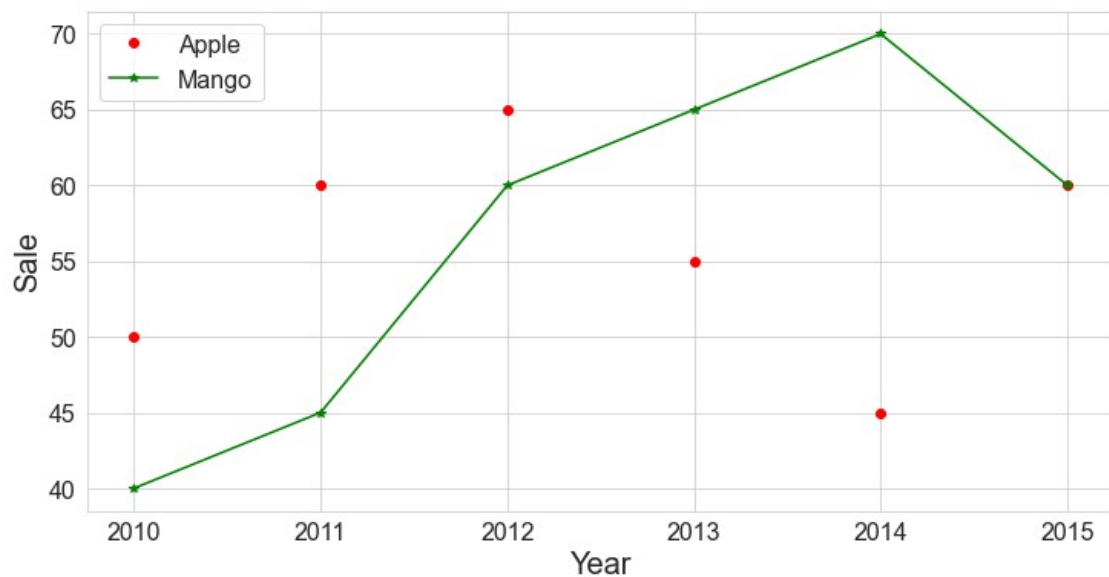
```
sns.set_style("darkgrid")
plt.figure(figsize=(12,6))
plt.plot(x,y,'or');#remove 1st line
plt.plot(x,y1,'*-g');
plt.xlabel("Year",size='20')
plt.ylabel("Sale",size='20')
plt.legend(['Apple','Mango'])
```

<matplotlib.legend.Legend at 0x1f0d912d670>

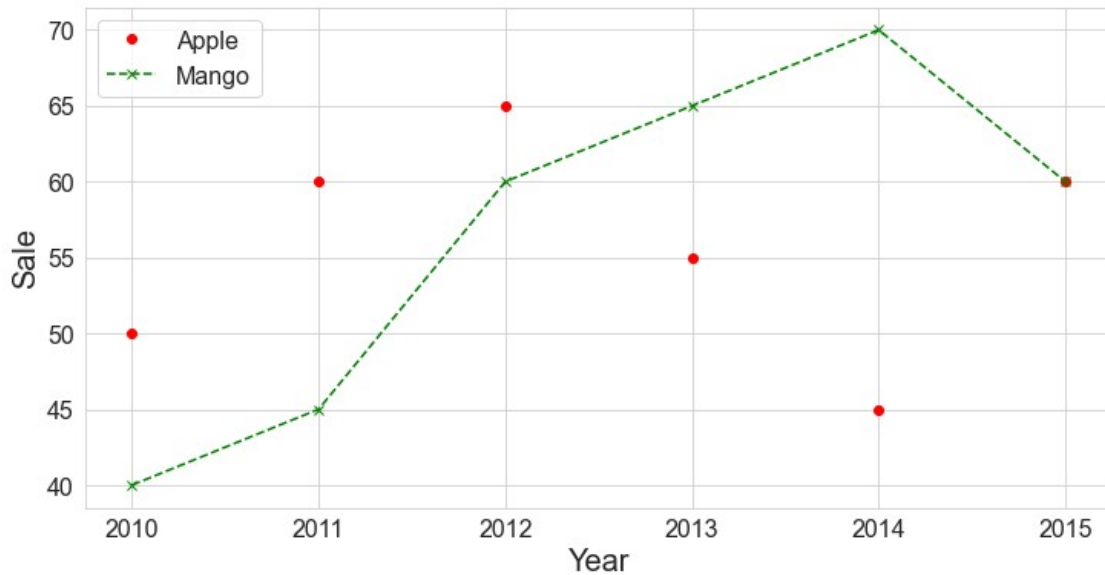


```
sns.set_style("whitegrid")
plt.figure(figsize=(12,6))
plt.plot(x,y,'or');#remove 1st line
plt.plot(x,y1,'*-g');
plt.xlabel("Year",size='20')
plt.ylabel("Sale",size='20')
plt.legend(['Apple','Mango'])
```

<matplotlib.legend.Legend at 0x1f0d8fcb790>



```
plt.figure(figsize=(12,6));
plt.plot(x,y,'or');#remove 1st line
plt.plot(x,y1,'x--g');
plt.xlabel("Year",size='20');
plt.ylabel("Sale",size='20');
plt.legend(['Apple','Mango']);
```



Scatter Plot

```
#load data
```

```
df = sns.load_dataset('iris')
```

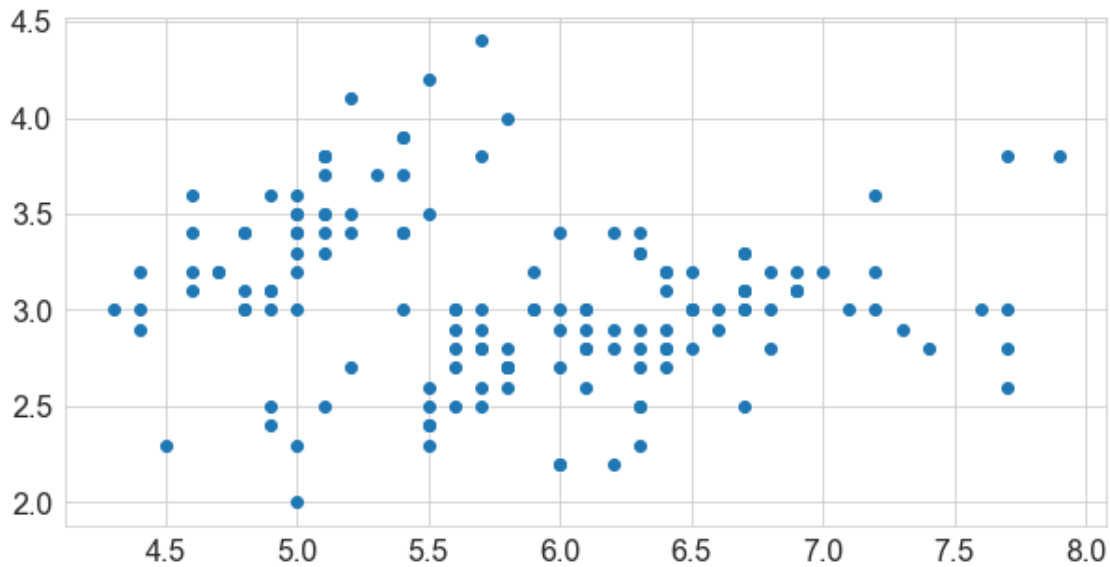
```
df['species'].unique()
```

```
array(['setosa', 'versicolor', 'virginica'], dtype=object)
```

```
df.head()
```

	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

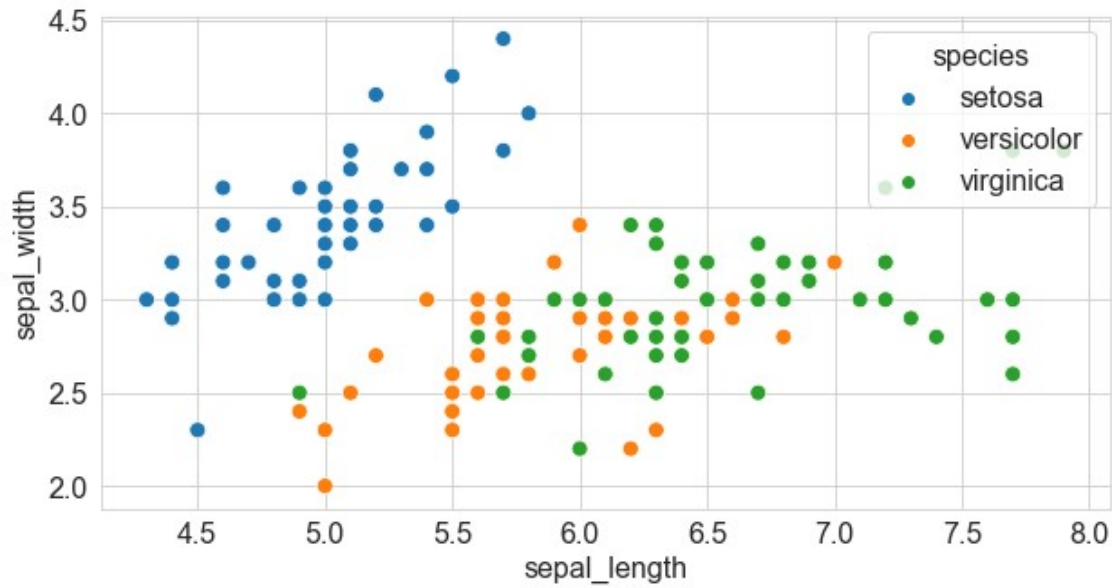
```
plt.scatter(df['sepal_length'],df['sepal_width']);
```



Adding Hues

```
sns.scatterplot(df['sepal_length'],df['sepal_width'],hue=df['species'],s=80);
```

C:\Program Files\Anaconda\lib\site-packages\seaborn_decorators.py:36:
FutureWarning: Pass the following variables as keyword args: x, y.
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(



Histogram

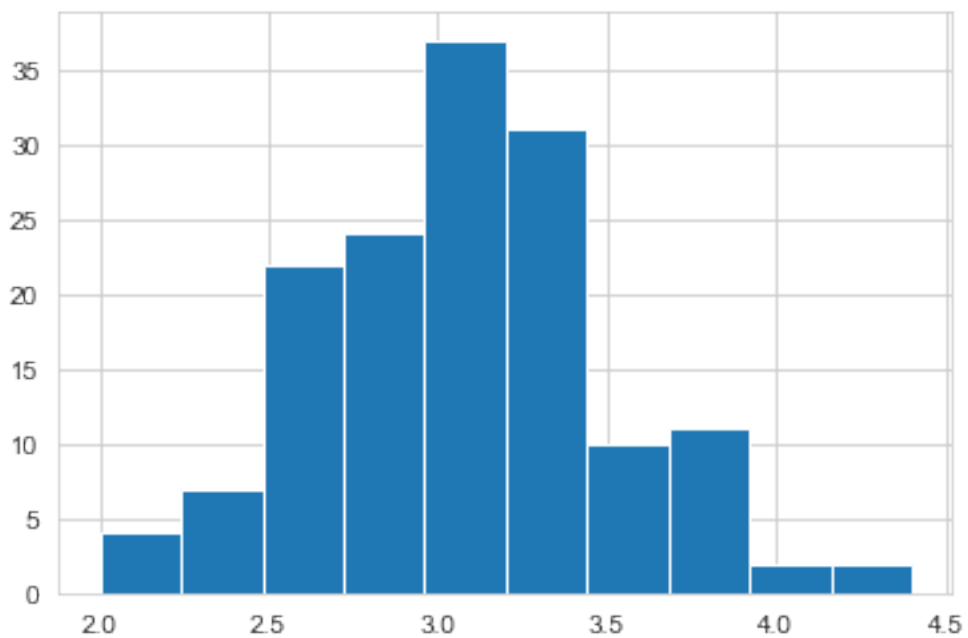
df.columns

```
Index(['sepal_length', 'sepal_width', 'petal_length', 'petal_width',
      'species'],
      dtype='object')
```

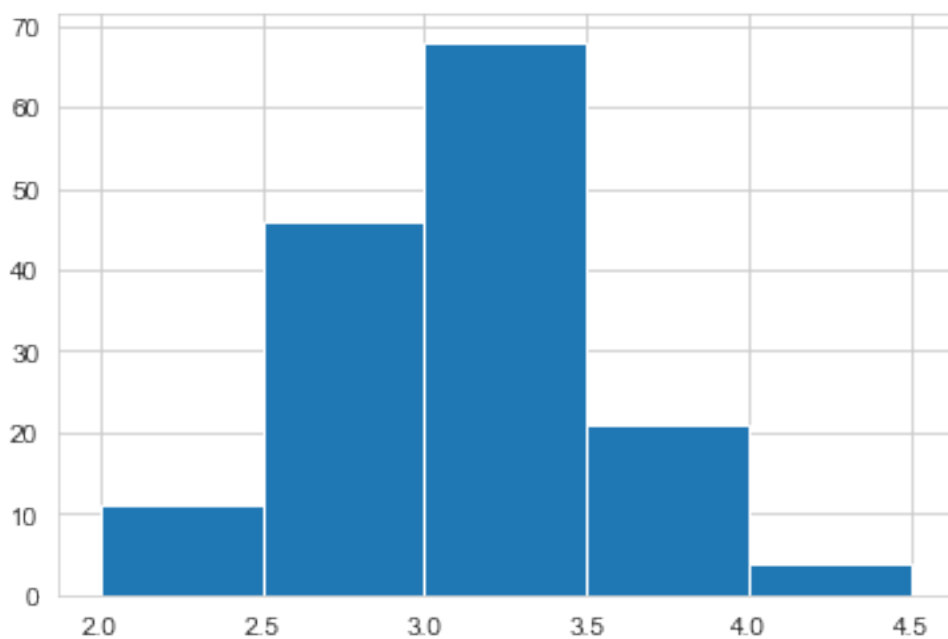
df.describe()

	sepal_length	sepal_width	petal_length	petal_width
count	150.000000	150.000000	150.000000	150.000000
mean	5.843333	3.057333	3.758000	1.199333
std	0.828066	0.435866	1.765298	0.762238
min	4.300000	2.000000	1.000000	0.100000
25%	5.100000	2.800000	1.600000	0.300000
50%	5.800000	3.000000	4.350000	1.300000
75%	6.400000	3.300000	5.100000	1.800000
max	7.900000	4.400000	6.900000	2.500000

```
plt.hist(df.sepal_width);
```



```
#hist use with bins  
val = [2,2.5,3,3.5,4,4.5]  
plt.hist(df.sepal_width,bins=val);
```



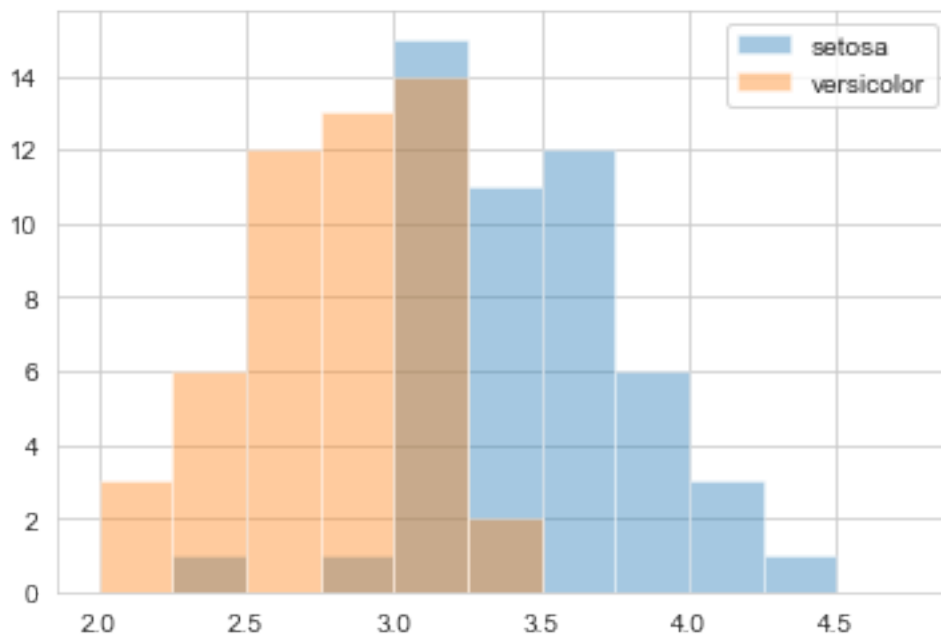
```
#check unique value in species  
list(df['species'].unique())  
['setosa', 'versicolor', 'virginica']
```

```
setosa = df[df['species']=='setosa']
versicolor=df[df['species']=='versicolor']
virginica=df[df['species']=='virginica']
```

plot 2 hist

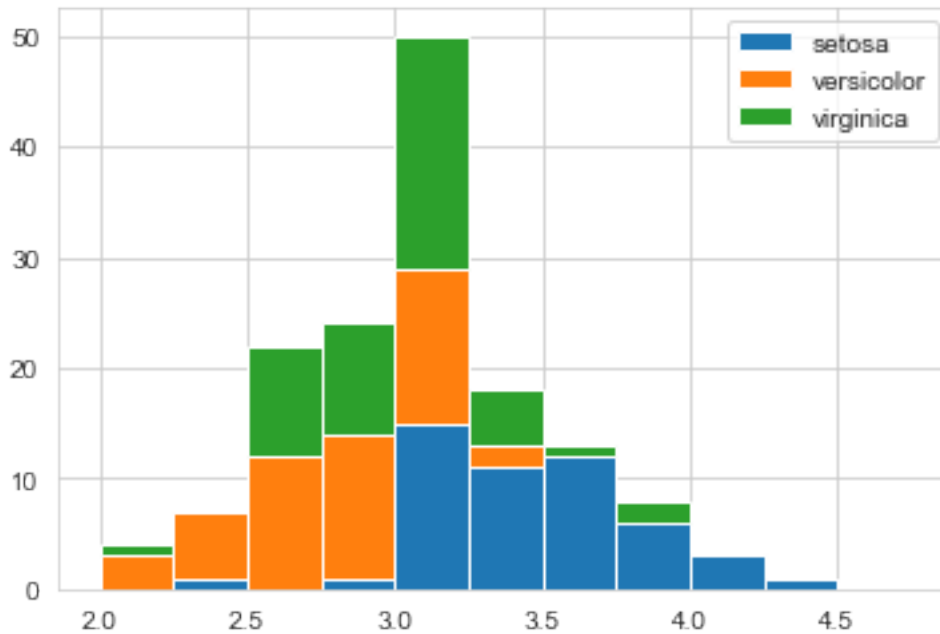
```
plt.hist(setosa['sepal_width'],alpha=0.4, bins=np.arange(2,5,0.25));
plt.hist(versicolor['sepal_width'],alpha=0.4,bins=np.arange(2,5,0.25))
;
plt.legend(['setosa','versicolor'])
```

<matplotlib.legend.Legend at 0x1f0dfe579a0>



Stacked histogram

```
plt.hist([setosa['sepal_width'],versicolor['sepal_width'],virginica['s
epal_width']],bins=np.arange(2,5,0.25),stacked=True);
plt.legend(['setosa','versicolor','virginica']);
```



BAR CHART

```
x=[2000,2001,2002,2003,2004,2005]
```

```
y = [10,14,12,16,9,13]
```

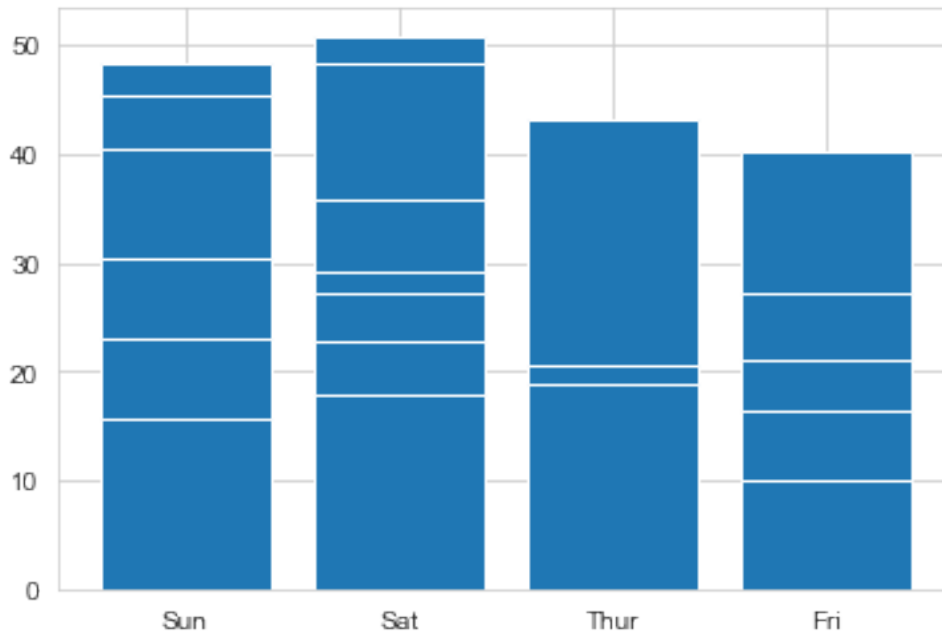
```
y1 = [12,16,15,13,17,18]
```

```
df1 = sns.load_dataset('tips')
```

```
df1.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

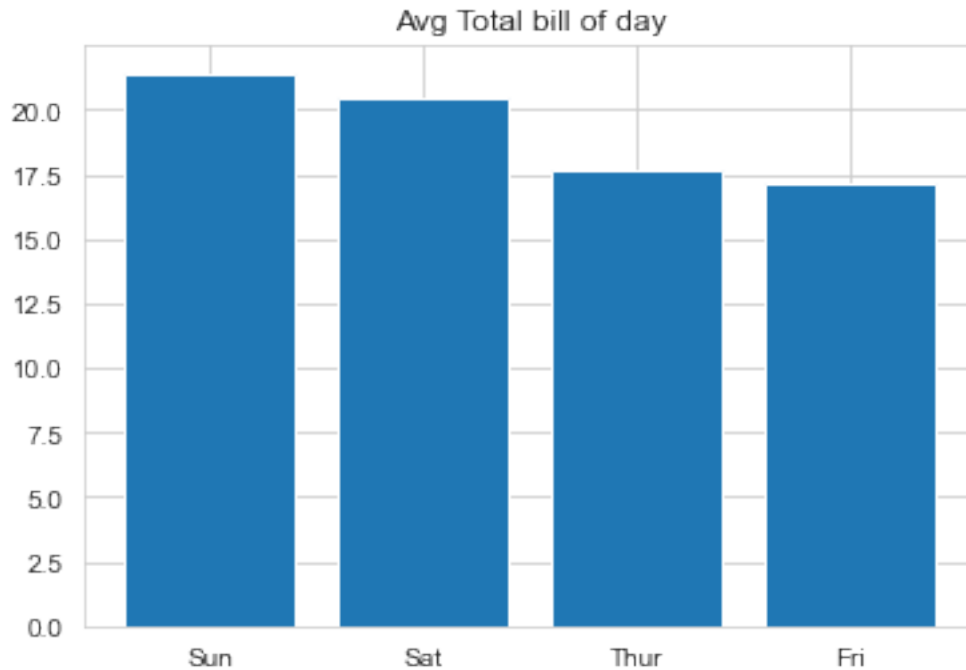
```
plt.bar(df1['day'],df1['total_bill']);
```



```
avg_total_bill = []
for i in list(df1['day'].unique()):
    avg_total_bill.append(round(np.average(df1[df1['day']==i]
['total_bill']),2))
avg_total_bill

[21.41, 20.44, 17.68, 17.15]

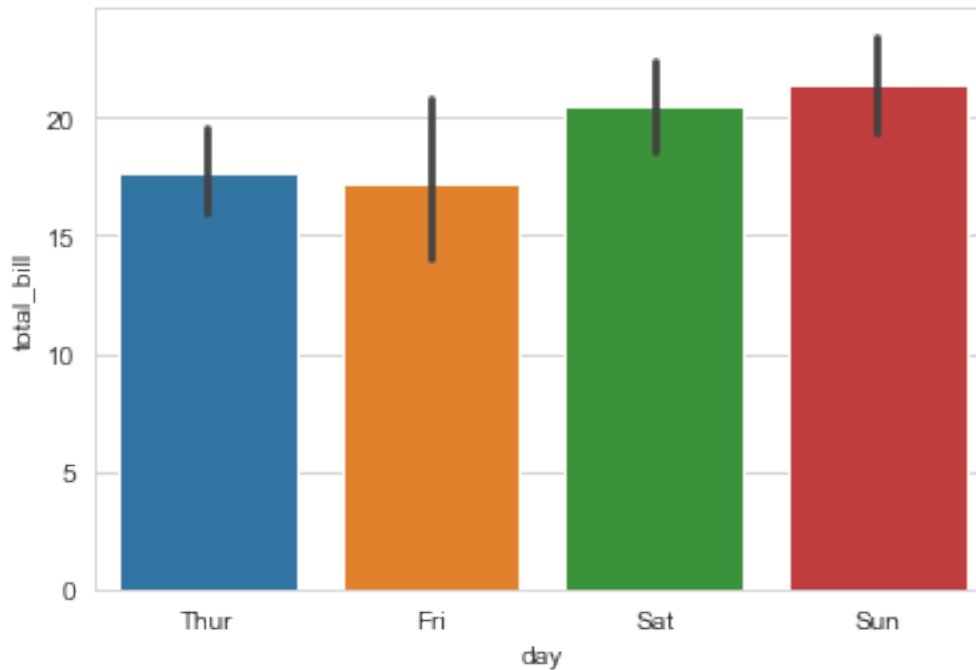
plt.bar(df1['day'].unique(),avg_total_bill);
plt.title("Avg Total bill of day");
```



Bar plot and average plot using of seaborn

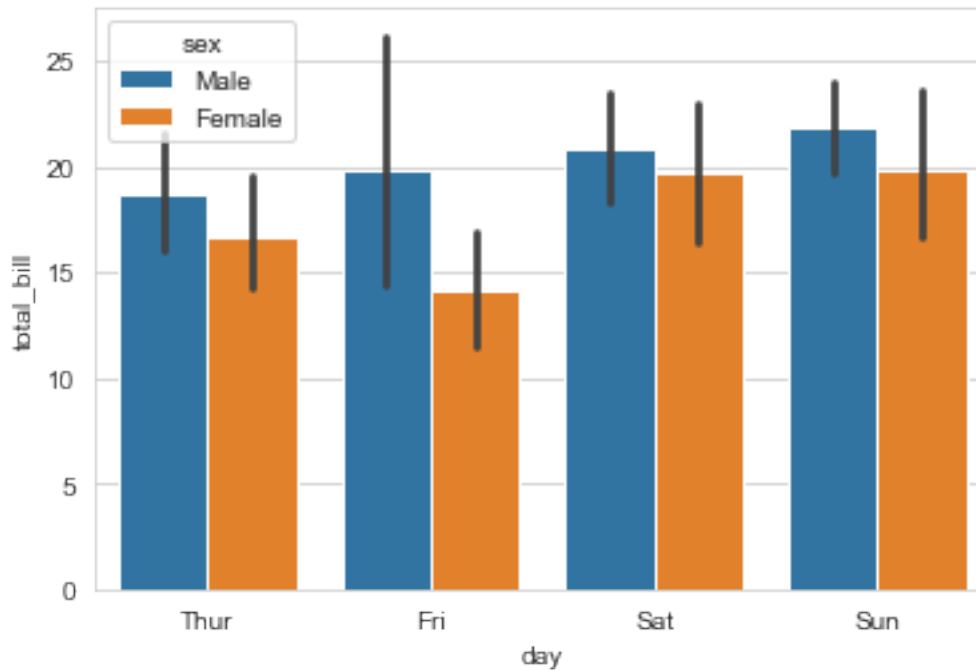
```
sns.barplot('day', 'total_bill', data=df1);
```

C:\Program Files\Anaconda\lib\site-packages\seaborn_decorators.py:36:
FutureWarning: Pass the following variables as keyword args: x, y.
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(



```
sns.barplot('day', 'total_bill', hue='sex', data=df1);
```

C:\Program Files\Anaconda\lib\site-packages\seaborn_decorators.py:36:
FutureWarning: Pass the following variables as keyword args: x, y.
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()



Heatmap

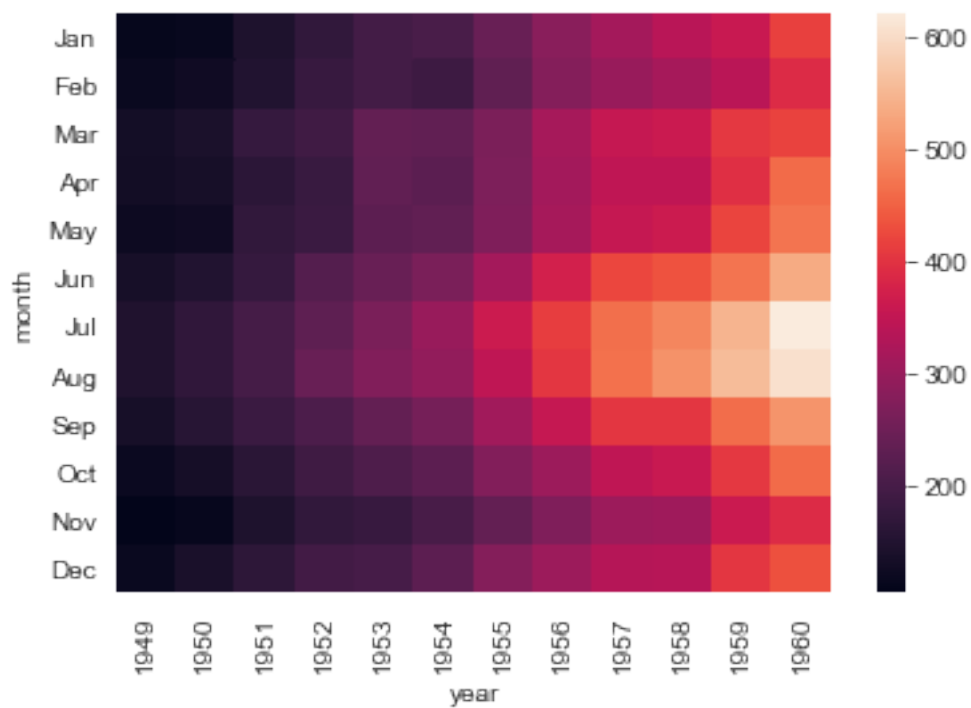
```
df2 = sns.load_dataset('flights').pivot('month', 'year', 'passengers')
```

df2

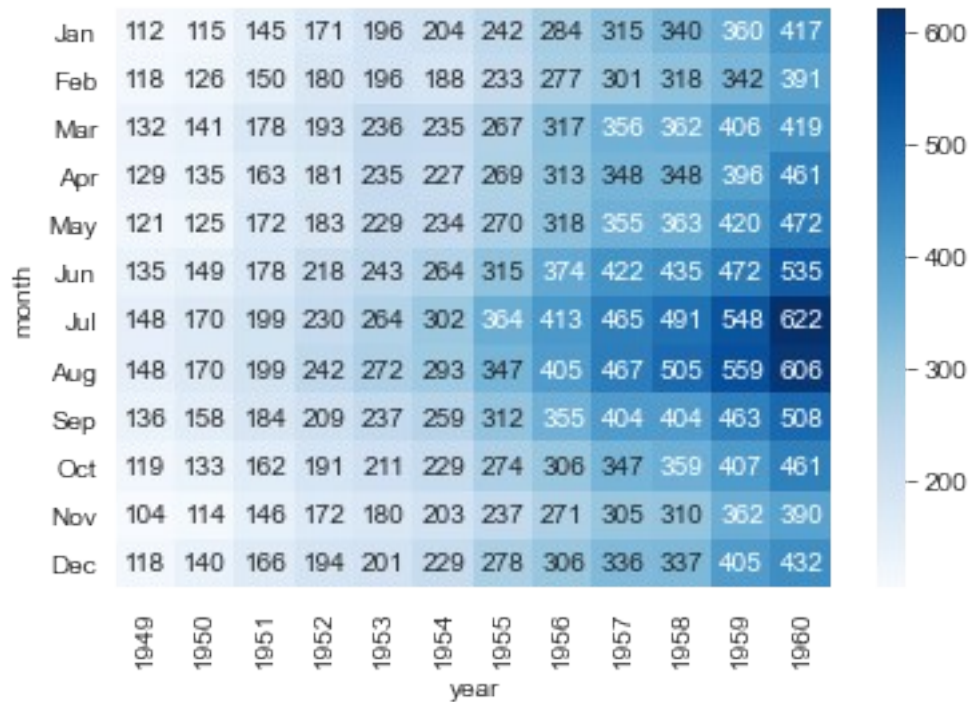
	1949	1950	1951	1952	1953	1954	1955	1956	1957	1958
year	1949	1950	1951	1952	1953	1954	1955	1956	1957	1958
1959	1960									
month										
Jan	112	115	145	171	196	204	242	284	315	340
360	417									
Feb	118	126	150	180	196	188	233	277	301	318
342	391									
Mar	132	141	178	193	236	235	267	317	356	362
406	419									
Apr	129	135	163	181	235	227	269	313	348	348
396	461									
May	121	125	172	183	229	234	270	318	355	363
420	472									
Jun	135	149	178	218	243	264	315	374	422	435
472	535									
Jul	148	170	199	230	264	302	364	413	465	491
548	622									
Aug	148	170	199	242	272	293	347	405	467	505

559	606										
Sep	136	158	184	209	237	259	312	355	404	404	
463	508										
Oct	119	133	162	191	211	229	274	306	347	359	
407	461										
Nov	104	114	146	172	180	203	237	271	305	310	
362	390										
Dec	118	140	166	194	201	229	278	306	336	337	
405	432										

```
sns.heatmap(df2);
```



```
sns.heatmap(df2,annot=True,cmap="Blues",fmt='d');
```



Images

```
from urllib.request import urlretrieve #its use for download image
from PIL import Image
```

```
url = r"C:\Users\goura\Downloads\1228946.jpg"
img = Image.open(url)
```

```
img_array = np.array(img)
```

```
img_array[500,2000]
```

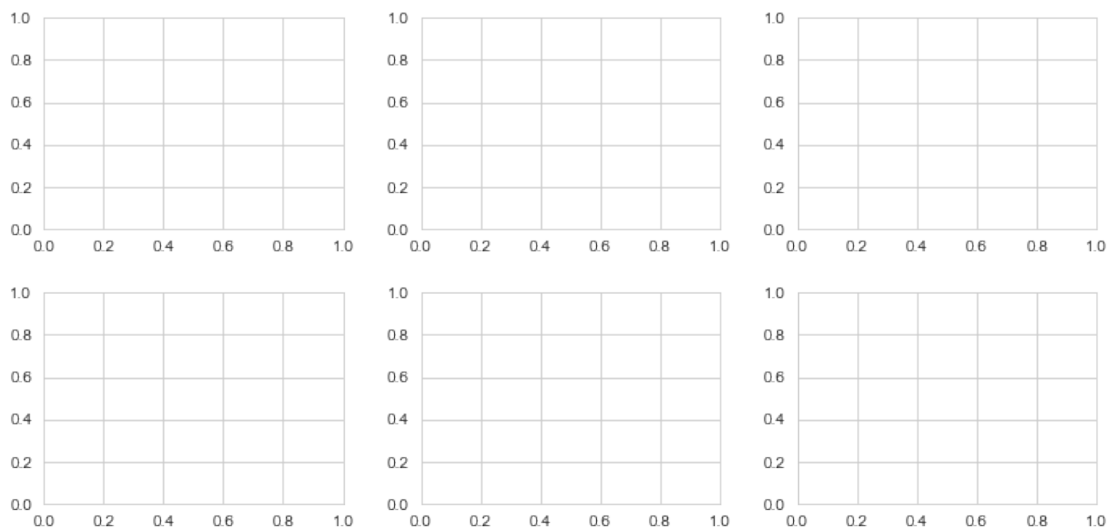
```
array([248, 169, 90], dtype=uint8)
```

```
plt.imshow(img_array);
```

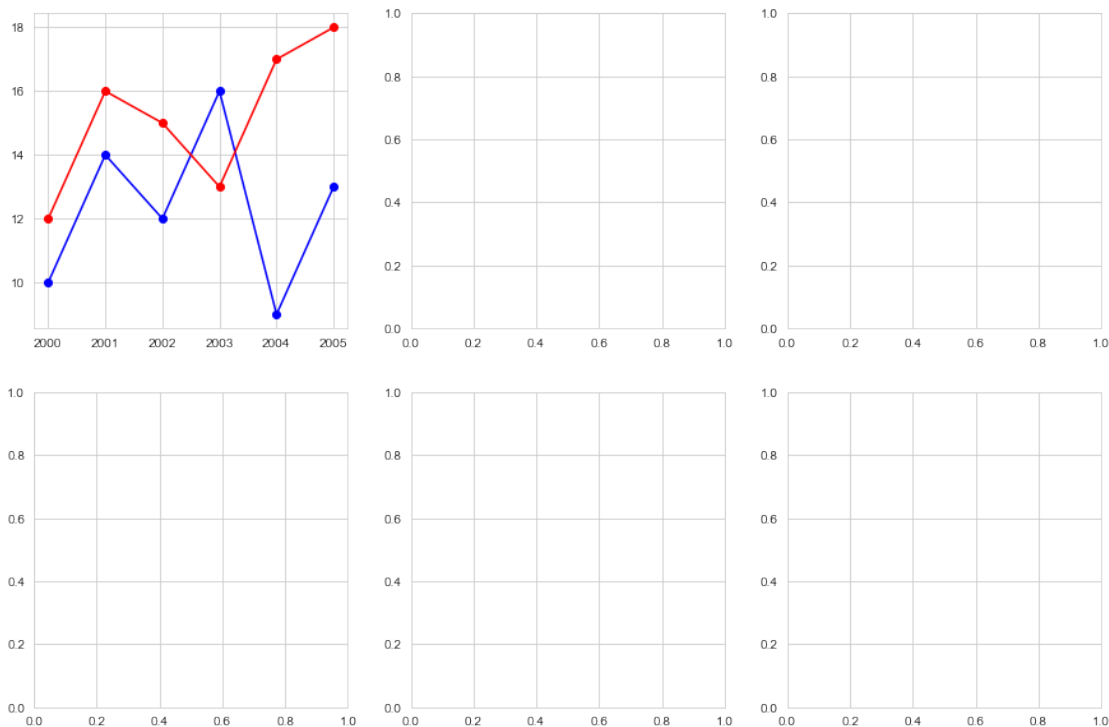


Plot multiple chart

```
plt.subplots(2,3,figsize=(10,5));  
plt.tight_layout(pad=2)#use for space for 1 graph to other graph
```



```
fig, axes = plt.subplots(2, 3, figsize=(15, 10));
axes[0, 0].plot(x, y, 'o-b');
axes[0, 0].plot(x, y1, 'o-r');
```



```
fig, axes = plt.subplots(2, 3, figsize=(15, 10));
axes[0, 0].plot(x, y, 'o-b');
axes[0, 0].plot(x, y1, 'o-r');
sns.scatterplot(df['sepal_length'], df['sepal_width'], hue=df['species'],
               , s=80, ax=axes[0, 1]);
sns.barplot('day', 'total_bill', hue='sex', data=df1, ax=axes[0, 2]);

axes[1, 0].hist([setosa['sepal_width'], versicolor['sepal_width'], virginica['sepal_width']], bins=np.arange(2, 5, 0.25), stacked=True);
axes[1, 0].legend(['setosa', 'versicolor', 'virginica']);
```

```
sns.heatmap(df2, annot=True, cmap="Blues", fmt='d', ax=axes[1, 1]);
axes[1, 2].imshow(img);
```

C:\Program Files\Anaconda\lib\site-packages\seaborn_decorators.py:36:
FutureWarning: Pass the following variables as keyword args: x, y.
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(
C:\Program Files\Anaconda\lib\site-packages\seaborn_decorators.py:36:
FutureWarning: Pass the following variables as keyword args: x, y.
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()

