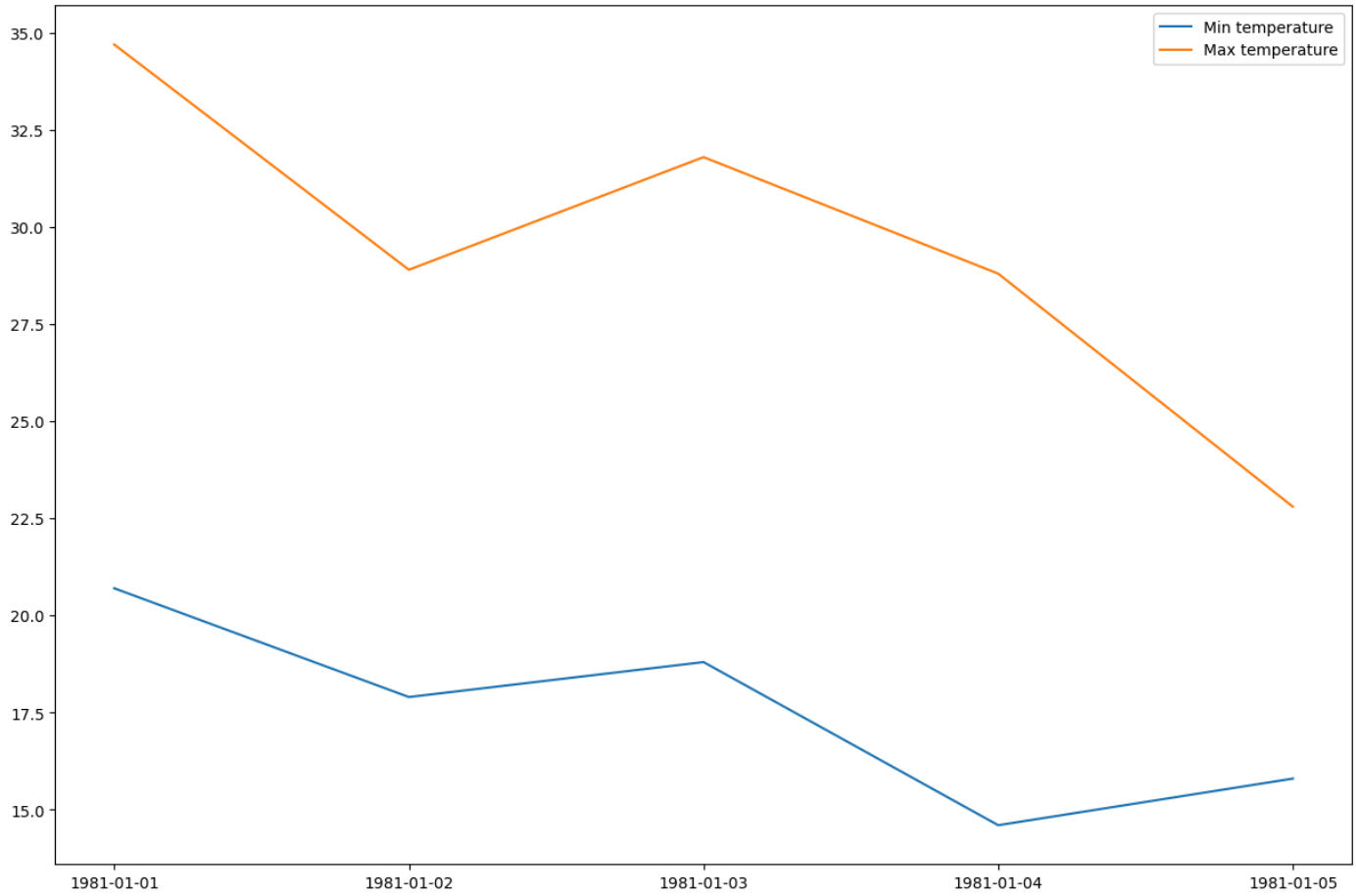
Double-click (or enter) to edit

```
import numpy as np
import pandas as pd
import matplotlib.pyplot as plt
from matplotlib.pyplot import figure
import seaborn as sns
%matplotlib inline
```

```
dates=['1981-01-01','1981-01-02','1981-01-03','1981-01-04','1981-01-05']
min_temperature=[20.7,17.9,18.8,14.6,15.8]
max_temperature=[34.7,28.9,31.8,28.8,22.8]
fig,axes=plt.subplots(nrows=1, ncols=1, figsize=(15,10))
axes.plot(dates,min_temperature,label='Min temperature')
axes.plot(dates,max_temperature,label='Max temperature')
axes.legend()
```

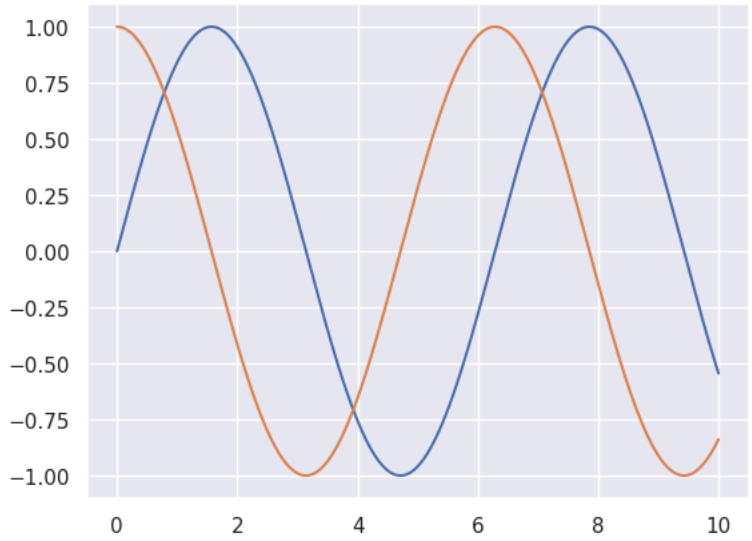
```
<matplotlib.legend.Legend at 0x7ec05641a980>
```




```
sns.set()
```

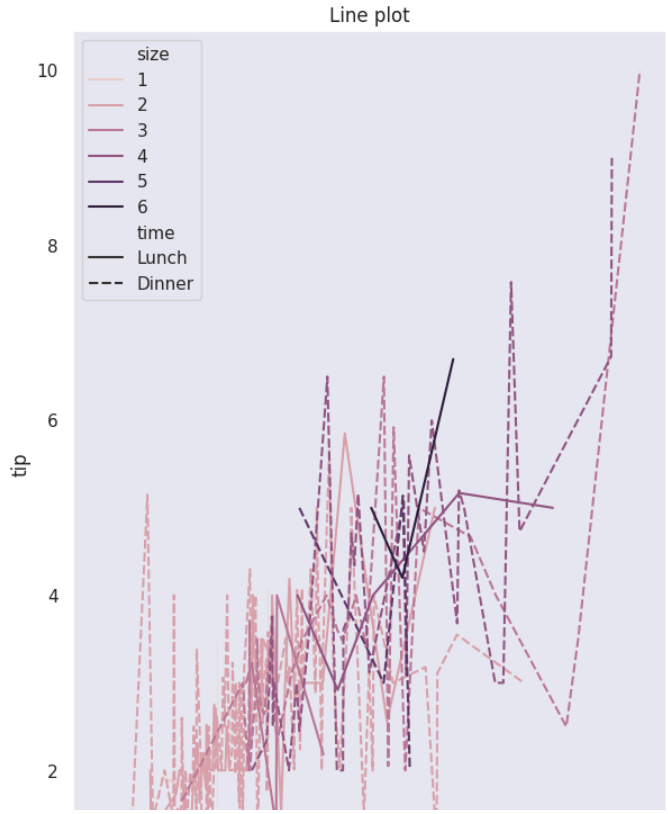
```
x=np.linspace(0, 10, 1000)
plt.plot(x, np.sin(x), x, np.cos(x))
```

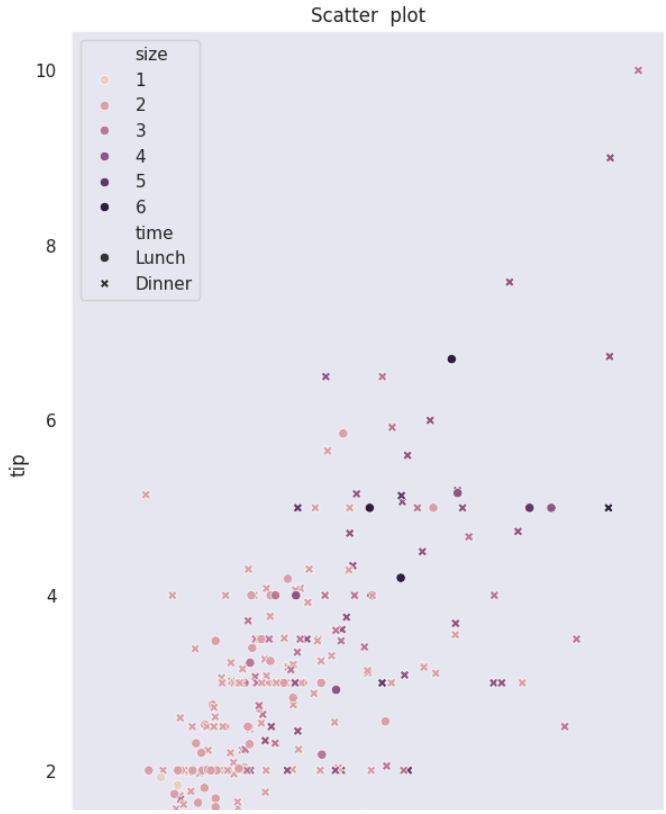
```
[<matplotlib.lines.Line2D at 0x7ec0561c4fa0>,
 <matplotlib.lines.Line2D at 0x7ec0561c5000>]
```




```
sns.set(style="dark")
fig, ax=plt.subplots(ncols=2, nrows=1 ,figsize=(15,10))
df=sns.load_dataset("tips")
print(df.head())

sns.lineplot(x="total_bill",y="tip",hue="size",style="time",data=df,ax=ax[0]).set_title("Line plot")
Sct_plt=sns.scatterplot(x="total_bill",y="tip",hue="size",style="time",data=df,ax=ax[1]).set_title("Scatter  plot")
Sct_plt.figure.savefig("Scatter_plot1.png")
print('Plot Saved')

sns.set_style('darkgrid')
fig, ax=plt.subplots(nrows=5,ncols=2)

sns.set_style('darkgrid')
fig, ax=plt.subplots(nrows=5,ncols=2)
fig.set_size_inches(18.5,10.5)
df=sns.load_dataset('tips')
sns.barplot(x='sex', y='total_bill',data=df,palette='plasma',estimator=np.std,ax=ax[0,0]).set_title('Bar plot')
sns.countplot(x='sex',data=df,ax=ax[0,1]).set_title('Count plot')
sns.boxplot(x='day',y='total_bill',data=df,hue='smoker',ax=ax[1,0]).set_title('Box plot')
sns.violinplot(x='day',y='total_bill',data=df,hue='sex',split=True,ax=ax[1,1]).set_title('Violin plot')
sns.stripplot(x='day',y='total_bill',data=df,jitter=True,hue='smoker',dodge=True,ax=ax[2,0]).set_title('Strip plot')
sns.swarmplot(x='day',y='total_bill',data=df,color='black',ax=ax[3,0]).set_title('Combined plot')
sns.boxenplot(x="day",y="total_bill",color="b",scale="linear",data=df,ax=ax[4,0])
sns.pointplot(x="day",y="total_bill",color="b",hue="sex",data=df,ax=ax[4,1])
sns.catplot(x="day",y="total_bill",data=df,kind="bar")
```

```
   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
Plot Saved
<ipython-input-7-0d6fef78d1f4>:18: FutureWarning:

Passing `palette` without assigning `hue` is deprecated and will be removed in v0.14.0. Assign the `x` variable to `hue` and set `legend

  sns.barplot(x='sex', y='total_bill',data=df,palette='plasma',estimator=np.std,ax=ax[0,0]).set_title('Bar plot')
<ipython-input-7-0d6fef78d1f4>:24: FutureWarning:

The `scale` parameter has been renamed to `width_method` and will be removed in v0.15. Pass `width_method='linear' for the same effect.
  sns.boxenplot(x="day",y="total_bill",color="b",scale="linear",data=df,ax=ax[4,0])
<ipython-input-7-0d6fef78d1f4>:25: FutureWarning:

Setting a gradient palette using color= is deprecated and will be removed in v0.14.0. Set `palette='dark:b'` for the same effect.

  sns.pointplot(x="day",y="total_bill",color="b",hue="sex",data=df,ax=ax[4,1])
<seaborn.axisgrid.FacetGrid at 0x7ec0561900d0>
```





```
# distribution plots is used for examining univariate and bivariate distributions 4 types are join,dist,pair,rug
sns.set_style('whitegrid')
df=sns.load_dataset('iris')
print(df.head())
sns.distplot(df['petal_length'],kde=True,color='red',bins=30).set_title('Dist Plot')
jointgrid=sns.JointGrid(x='petal_length',y='petal_width',data=df)
jointgrid.plot_joint(sns.scatterplot)

jointgrid.plot_marginals(sns.distplot)
g=sns.jointplot(x='petal_length',y='petal_width',data=df,kind='hex')
g.fig.suptitle('JointPlot')
g.fig.suptitle("Pair Plot 1")
pairgrid=sns.PairGrid(data=df)
pairgrid=pairgrid.map_offdiag(sns.scatterplot)
pairgrid=sns.PairGrid(data=df)
pairgrid=pairgrid.map_upper(sns.scatterplot)
pairgrid=pairgrid.map_diag(plt.hist)
pairgrid=pairgrid.map_lower(sns.kdeplot)
g=sns.PairGrid(df,diag_sharey=False,corner=True)
g.map_lower(sns.scatterplot)
g.map_diag(sns.kdeplot)
fig, ax=plt.subplots(nrows=2,ncols=2,figsize=(15,10))
df1=sns.load_dataset('flights')
df2=sns.load_dataset('iris')
df1=pd.pivot_table(values='passengers',index='month',columns='year',data=df1)
```

```
fig, ax=plt.subplots(nrows=2, ncols=2, figsize=(15,10))
df1 = sns.load_dataset('flights')
df2=sns.load_dataset('iris')
df11 = pd.pivot_table(values = 'passengers', index ='month', columns ='year', data = df1)
dfc1 = df1.corr(numeric_only=True)
dfc2 = df2.corr(numeric_only=True)
sns.heatmap(df11,cmap='YlGnBu', linecolor = 'r', linewidths = 0.5,annot=True,fmt='d',square=True,ax=ax[0,0]).set_title('Heat Map Flights')
sns.heatmap(dfc2,cmap='coolwarm', linecolor = 'black', linewidths = 1, annot=True,ax=ax[0,1]).set_title('Heat Map Iris')
mask1=np.triu(dfc2)
sns.heatmap(dfc2,annot=True,mask=mask1,ax=ax[1,0],cmap='coolwarm').set_title('Heat map Lower Triangle')
mask2=np.tril(dfc2)
sns.heatmap(dfc2,annot=True,cmap='Y1GnBu',mask=mask2,ax=ax[1,1]).set_title('Heat Upper Triangle')
mask2=np.tril(dfc2)
sns.clustermap(df11,cmap='RdY1Gn')
sns.clustermap(df11,cmap='plasma',standard_scale=1)
```

```
---------------------------------------------------------------------------
NameError                                 Traceback (most recent call last)
<ipython-input-1-20eb948cd6ef> in <cell line: 1>()
----> 1 fig, ax=plt.subplots(nrows=2, ncols=2, figsize=(15,10))
      2 df1 = sns.load_dataset('flights')
      3 df2=sns.load_dataset('iris')
```