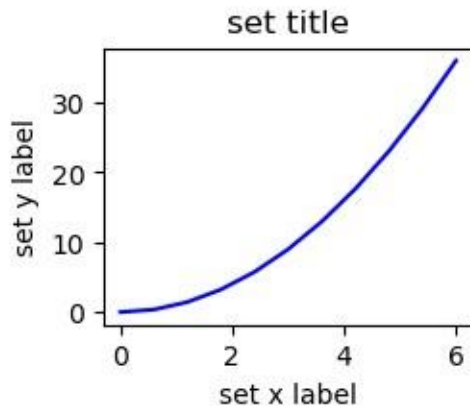


# matplotlib-seaborn

April 8, 2024

```
[1]: import numpy as np
import pandas as pd
import matplotlib.pyplot as plt
x=np.linspace(0,6,11)
y=x**2
fig=plt.figure()#create figure(empty canvas)
#plot on that set of axes
axes=fig.add_axes([0.01,0.01,0.3,0.3])#left,bottom,width,height(range 0 to 1)
axes.plot(x,y,'b')
#notice the use of set_ to begin methods
axes.set_xlabel("set x label")
axes.set_ylabel("set y label")
axes.set_title("set title")
```

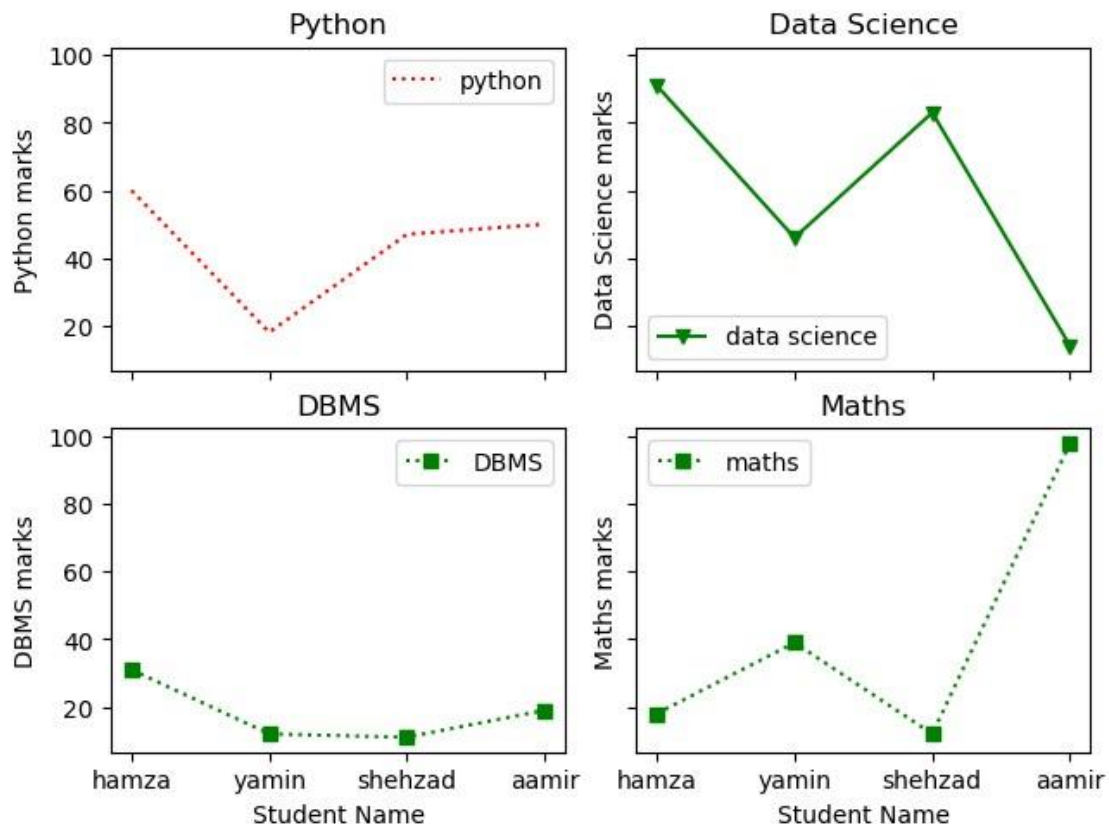
```
[1]: Text(0.5, 1.0, 'set title')
```



```
[2]: df=pd.DataFrame(np.random.
    randint(0,100,(4,4)),index=["karnavi","diya","reeva","james"],columns=["python",
    "data_scienc df
fig2,ax=plt.subplots(2,2,sharex=True,sharey=True,layout="constrained")
ax[0,0].plot(df.index,df.python,'r',ls="dotted",label="python")
ax[0,0].set_ylabel("Python marks")
ax[0,0].set_title("Python") ax[0,0].legend()
ax[0,1].plot(df.index,df.data_science,'gv-',label="data
science") ax[0,1].set_ylabel("Data Science marks")
ax[0,1].set_title("Data Science") ax[0,1].legend(loc=3)
```

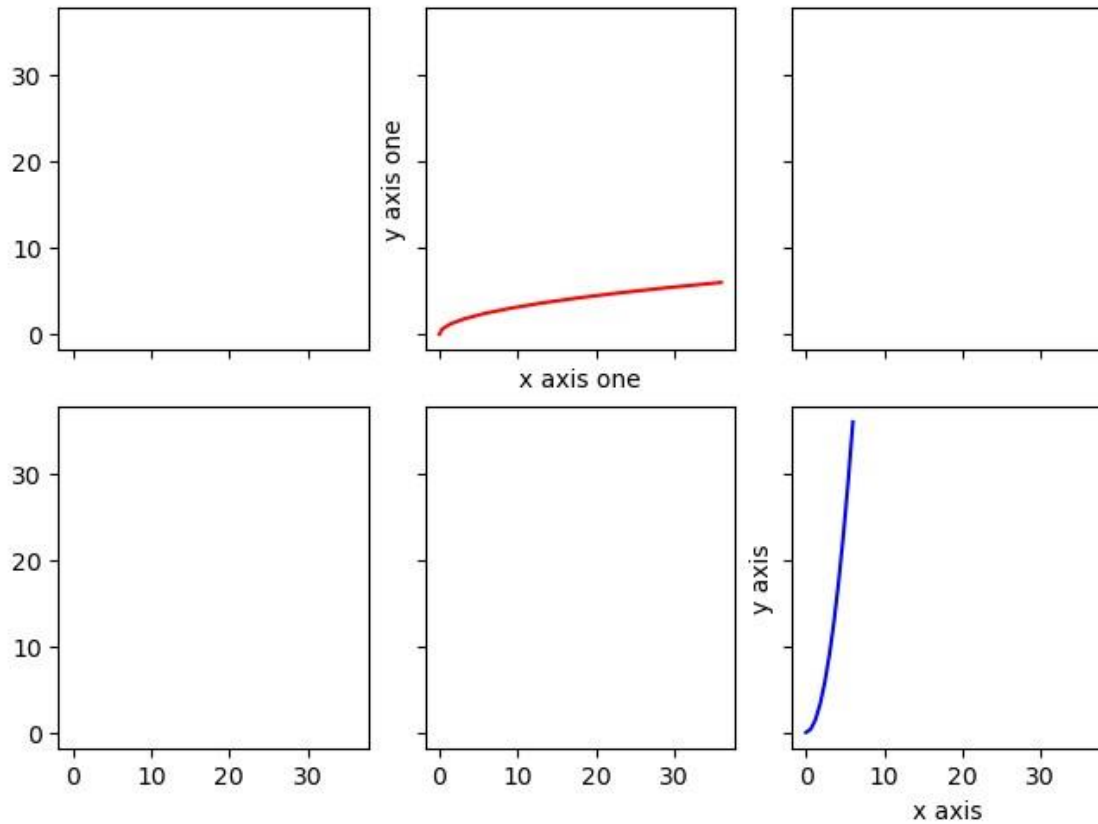
```
ax[1,0].plot(df.index,df.DBMS,'gs',linestyle="dotted",label="DBMS") ax[1,0].set_ylabel("DBMS marks")
ax[1,0].set_xlabel("Student Name")
ax[1,0].set_title("DBMS") ax[1,0].legend()
ax[1,1].plot(df.index,df.Maths,'gs',linestyle="dotted",label="maths") ax[1,1].set_xlabel("Student Name")
ax[1,1].set_ylabel("Maths marks")
ax[1,1].set_title("Maths") ax[1,1].legend()
```

[2]: <matplotlib.legend.Legend at 0x249dc2720d0>



```
[3]: fig1,ax=plt.subplots(2,3,sharex=True,sharey=True,layout="constrained") ax[1,2].plot(x,y,'b') ax[0,1].plot(y,x,'r')
ax[1,2].set_xlabel("x axis") ax[1,2].set_ylabel("y axis")
ax[0,1].set_xlabel("x axis one") ax[0,1].set_ylabel("y axis one")
```

[3]: Text(0, 0.5, 'y axis one')

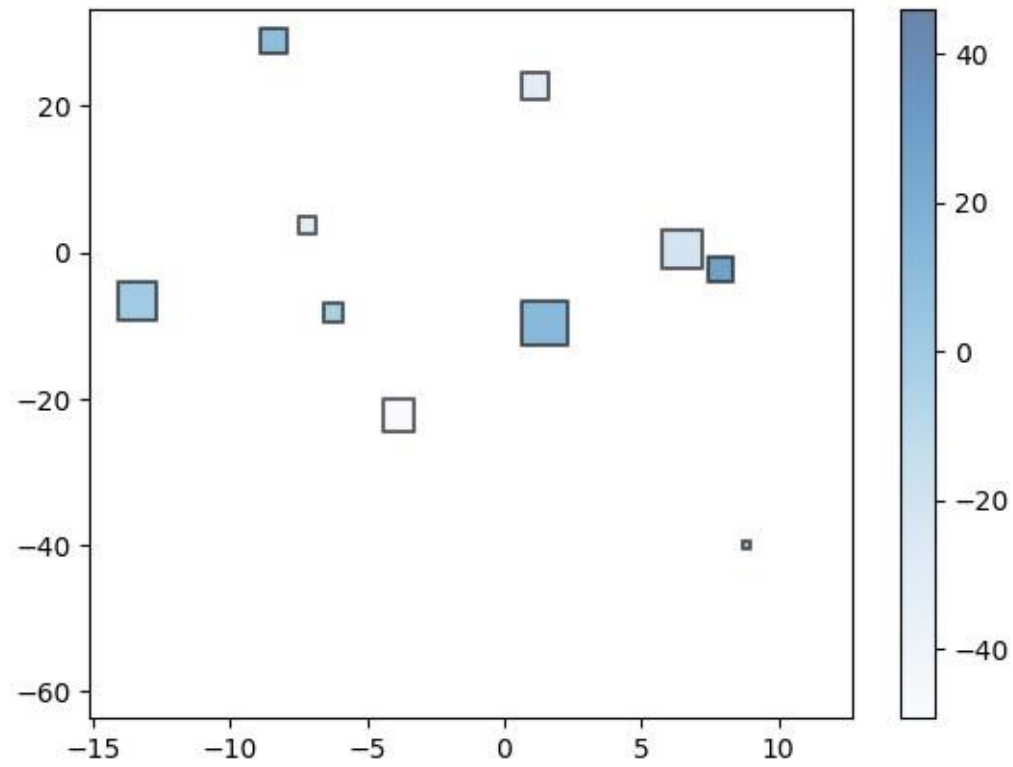


```
[4]: x=np.random.normal(0,10,20)
      y=np.random.normal(0,20,20)
      s1=np.random.normal(50,200,20)
      c1=np.random.normal(0,20,20)
```

```
[5]: plt.scatter(x,y,s=s1,marker='s',c=c1,alpha=0.
      ↵6,cmap='Blues',edgecolor='k',linewidth=1.5)
      plt.colorbar()
```

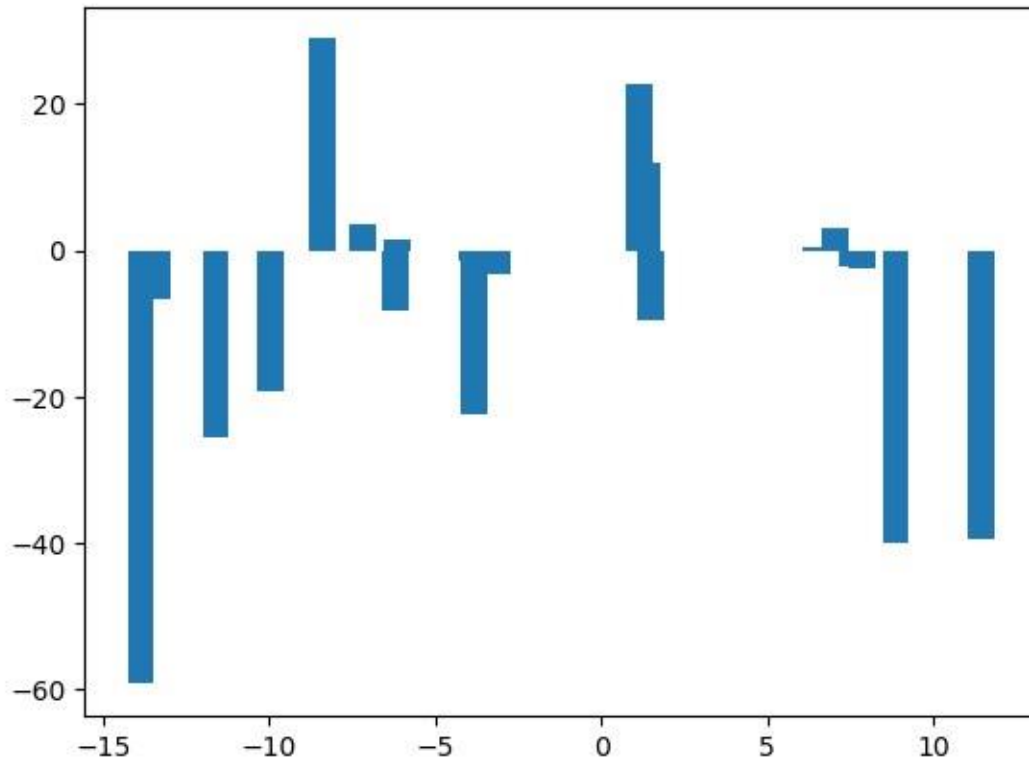
```
C:\Users\HP\anaconda3\Lib\site-packages\matplotlib\collections.py:963:
RuntimeWarning: invalid value encountered in sqrt
      scale = np.sqrt(self._sizes) * dpi / 72.0 *
              self._factor
```

```
[5]: <matplotlib.colorbar.Colorbar at 0x249dcb65810>
```



```
[6]: plt.bar(x,y)
```

```
[6]: <BarContainer object of 20 artists>
```



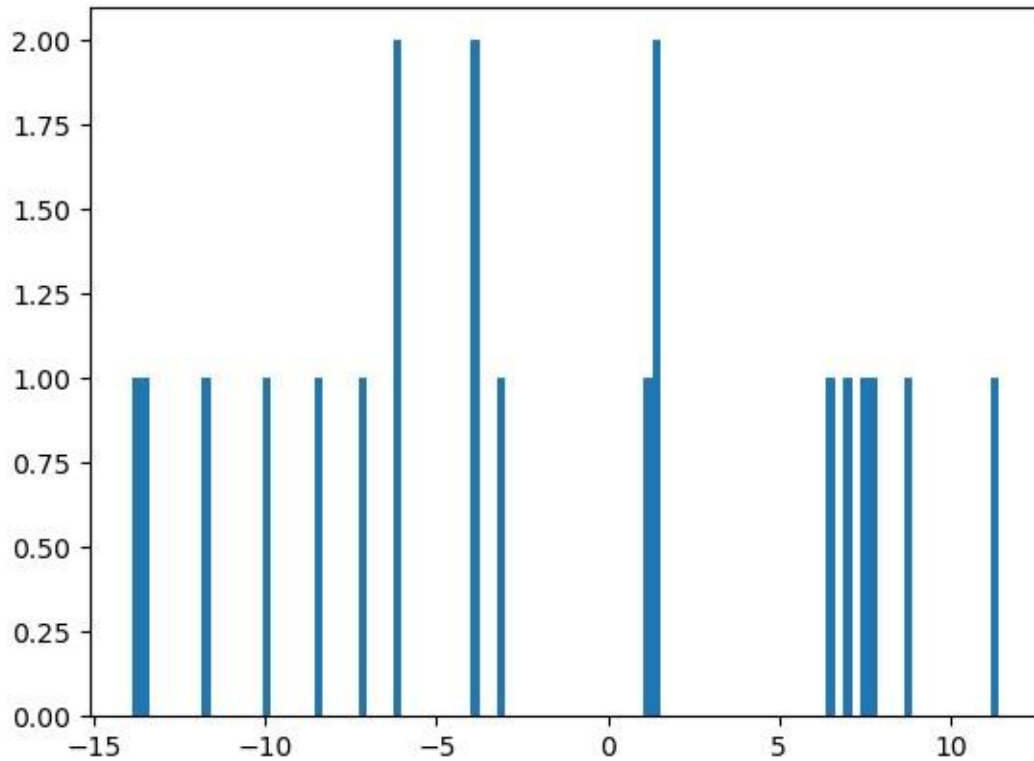
```
[7]: plt.hist(x,bins=100)
```

```
[7]: (array([1., 1., 0., 0., 0., 0., 0., 0., 1., 0., 0., 0., 0., 0., 0., 0.,
1., 0.,
0., 0., 0., 0., 1., 0., 0., 0., 0., 1., 0., 0., 0., 2., 0.,
0., 0., 0., 0., 0., 0., 0., 2., 0., 0., 1., 0., 0., 0., 0.,
0., 0., 0., 0., 0., 0., 0., 0., 0., 0., 0., 1., 2., 0.,
0., 0., 0., 0., 0., 0., 0., 0., 0., 0., 0., 0., 0., 0.,
0., 0., 0., 0., 0., 0., 0., 0., 0., 0., 0., 0., 1., 0., 1., 0.,
1.,
1., 0., 0., 0., 1., 0., 0., 0., 0., 0., 0., 0., 0., 0., 0., 1.]),
array([-13.85190231, -13.59929988, -13.34669746, -
13.09409503, -12.84149261, -12.58889019, -12.33628776, -
12.08368534,
-11.83108291, -11.57848049, -11.32587806, -11.07327564,
-10.82067321, -10.56807079, -10.31546836, -10.06286594,
-9.81026352, -9.55766109, -9.30505867, -9.05245624,
-8.79985382, -8.54725139, -8.29464897, -8.04204654,
-7.78944412, -7.53684169, -7.28423927, -7.03163685,
-6.77903442, -6.526432 , -6.27382957, -6.02122715,
-5.76862472, -5.5160223 , -5.26341987, -5.01081745,
-4.75821502, -4.5056126 , -4.25301018, -4.00040775,
```

```

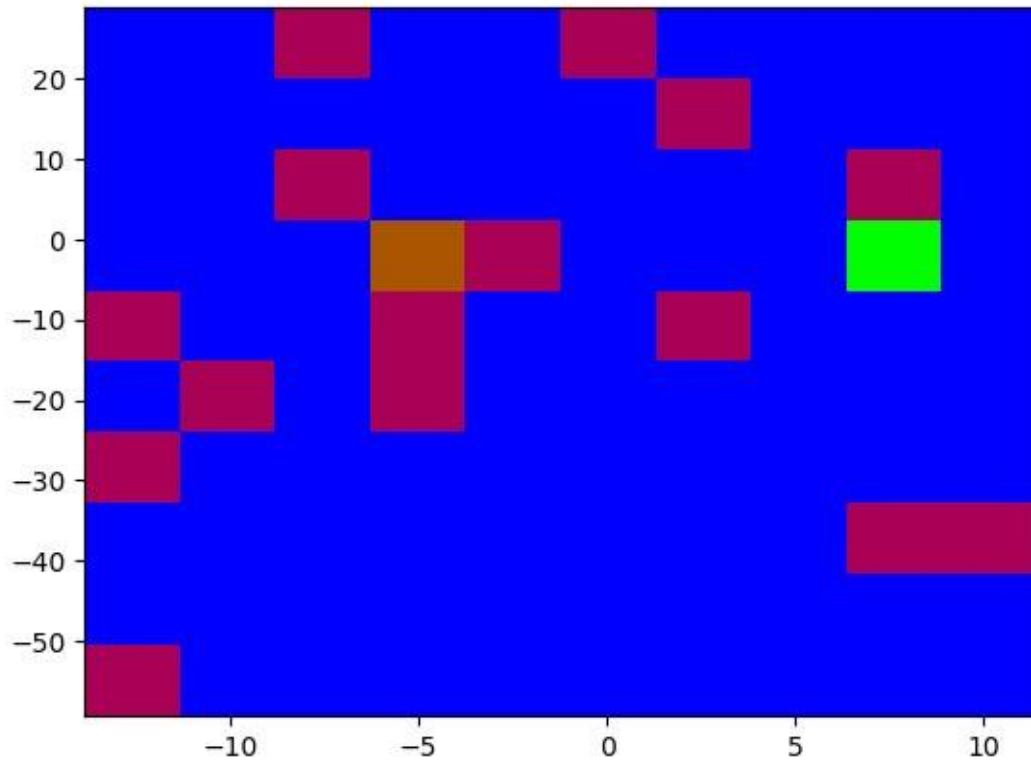
-3.74780533, -3.4952029 , -3.24260048, -2.98999805,
-2.73739563, -2.4847932 , -2.23219078, -1.97958835,
-1.72698593, -1.47438351, -1.22178108, -0.96917866,
-0.71657623, -0.46397381, -0.21137138, 0.04123104,
  0.29383347,          0.79903832, 1.05164074,
  0.54643589,
  1.30424316,          1.80944801, 2.06205044,
  1.55684559,
  2.31465286,          2.81985771, 3.07246014,
  2.56725529,
  3.32506256,          3.83026741, 4.08286983,
  3.57766499,
  4.33547226,          4.84067711, 5.09327953,
  4.58807468,
  5.34588196,          5.85108681, 6.10368923,
  5.59848438,
  6.35629166,          6.8614965 , 7.11409893,
  6.60889408,
  7.36670135,          7.8719062 , 8.12450863,
  7.61930378,
  8.37711105,          8.8823159 , 9.13491833,
  8.62971348,
  9.38752075,          9.8927256          ,
  9.64012317,          10.14532802,
10.39793045, 10.65053287, 10.9031353 , 11.15573772,
  11.40834015]),
<BarContainer object of 100 artists>)

```



```
[8]: plt.hist2d(x,y,bins=10,cmap='brg')
```

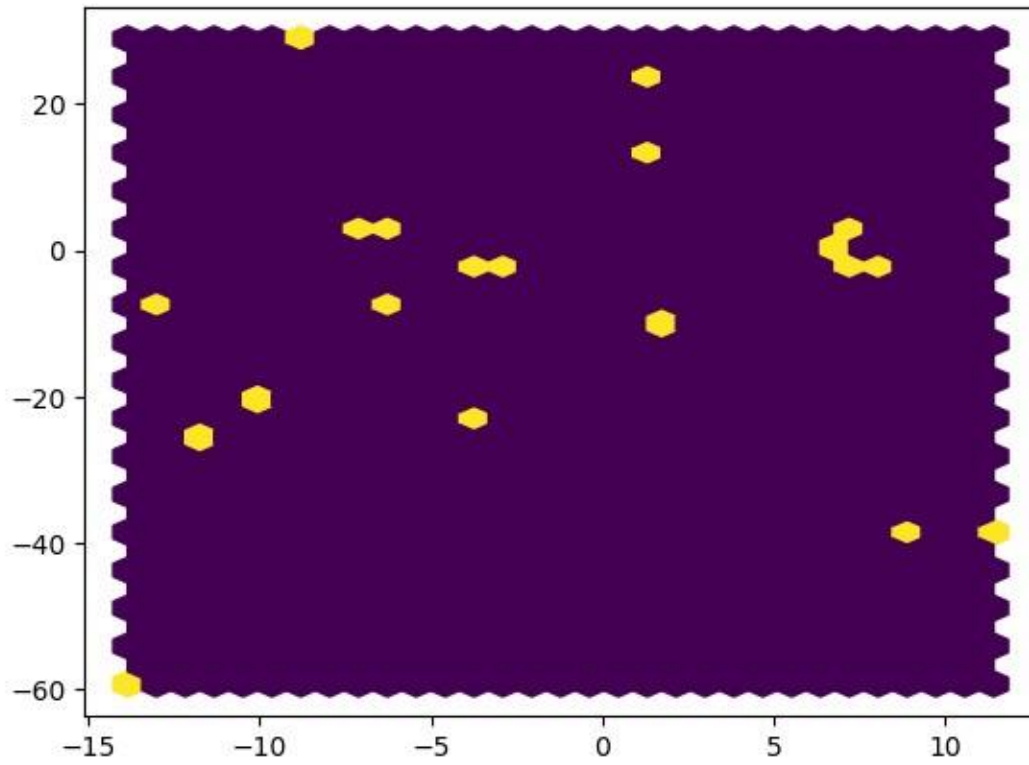
```
[8]: (array([[1., 0., 0., 1., 0., 1., 0., 0., 0.,
            0.], [0., 0., 0., 0., 1., 0., 0., 0., 0.,
            0., 0.],
            [0., 0., 0., 0., 0., 0., 0., 1., 0., 1.],
            [0., 0., 0., 0., 1., 1., 2., 0., 0., 0.],
            [0., 0., 0., 0., 0., 0., 1., 0., 0., 0.],
            [0., 0., 0., 0., 0., 0., 0., 0., 0., 1.],
            [0., 0., 0., 0., 0., 1., 0., 0., 1., 0.],
            [0., 0., 0., 0., 0., 0., 0., 0., 0., 0.],
            [0., 0., 1., 0., 0., 0., 3., 1., 0., 0.],
            [0., 0., 1., 0., 0., 0., 0., 0., 0., 0.]]), array([-
13.85190231, -11.32587806, -8.79985382, -6.27382957,
            -3.74780533, -1.22178108, 1.30424316, 3.83026741,
            6.35629166, 8.8823159 , 11.40834015]), array([-
59.34624118, -50.51880164, -41.69136209, -32.86392255,
            -24.03648301, -15.20904347, -6.38160393, 2.44583561,
            11.27327516, 20.1007147 , 28.92815424])),
<matplotlib.collections.QuadMesh at 0x249dcd46810>)
```



```
[9]: plt.hexbin(x,y,gridsize=30)
```

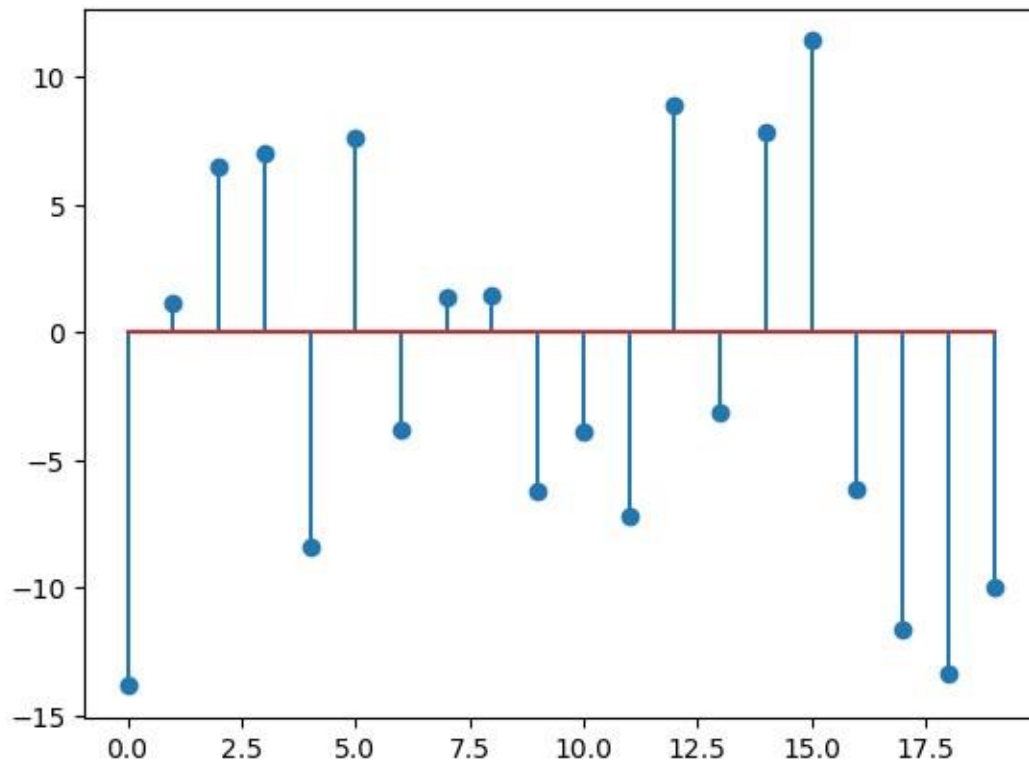
```
[9]: <matplotlib.collections.PolyCollection at 0x249dd0c9110>
```





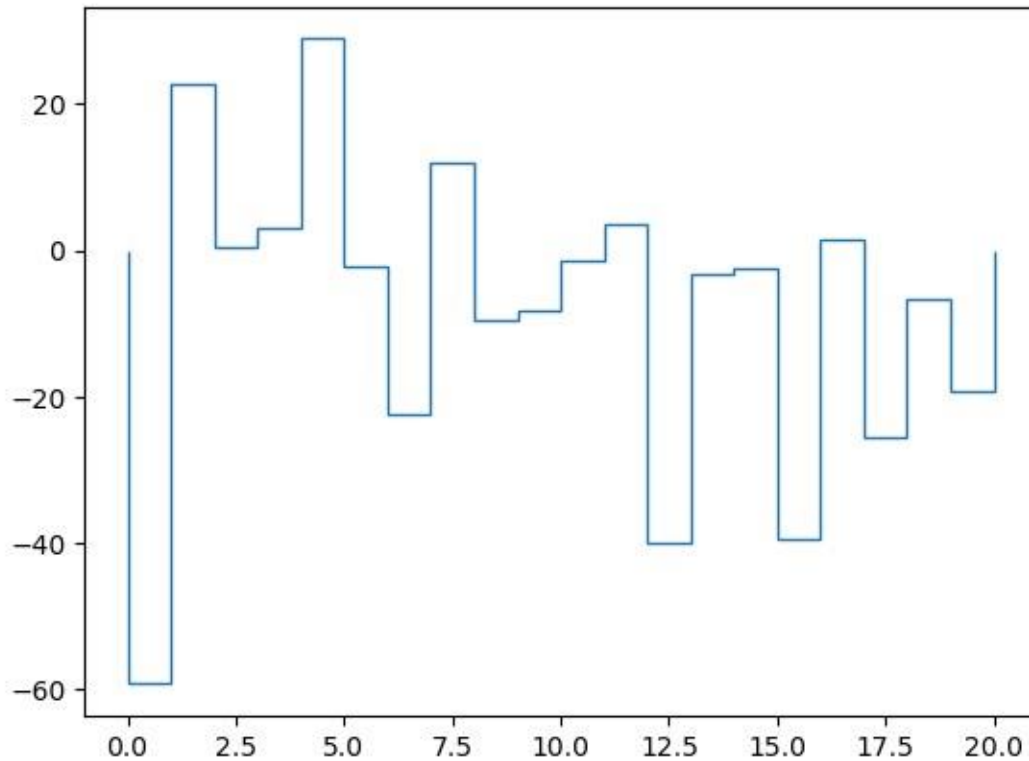
```
[10]: plt.stem(x)
```

```
[10]: <StemContainer object of 3 artists>
```



```
[11]: plt.stairs(y)
```

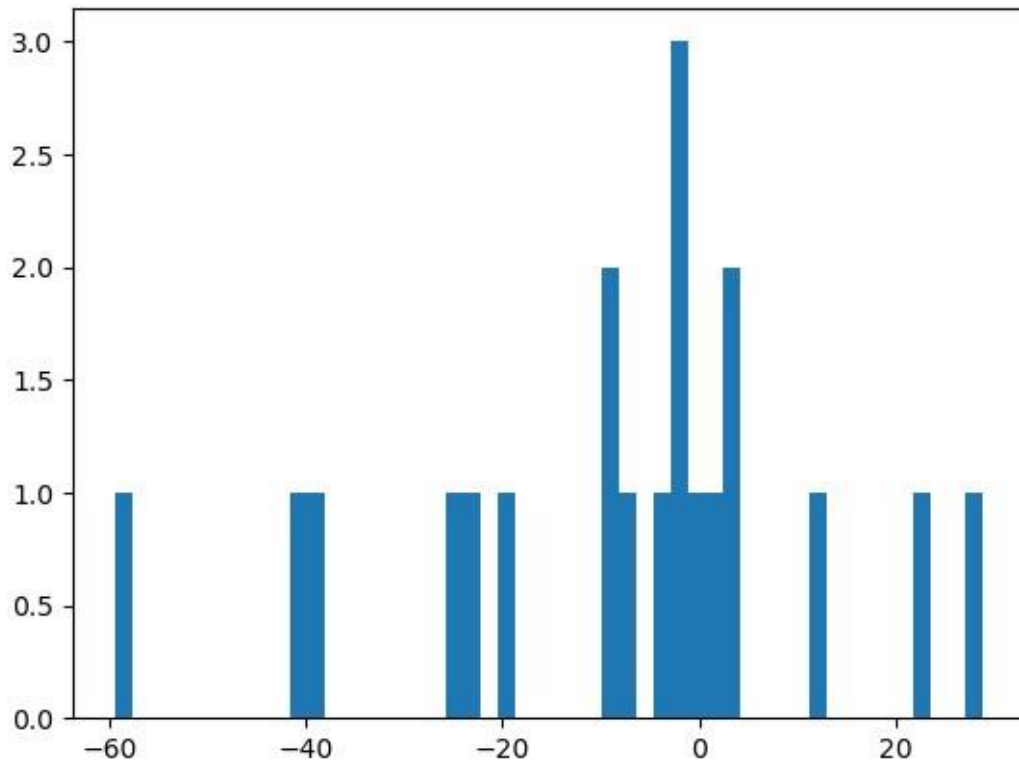
```
[11]: <matplotlib.patches.StepPatch at 0x249dd485590>
```



```
[13]: plt.hist(y,bins=50,histtype='stepfilled')
```

```
[13]: (array([1., 0., 0., 0., 0., 0., 0., 0., 0., 0., 0., 1., 1., 0., 0., 0.,
0., 0.,
          0., 0., 1., 1., 0., 1., 0., 0., 0., 0., 0., 2., 1., 0., 1., 3.,
1.,
          1., 2., 0., 0., 0., 0., 1., 0., 0., 0., 0., 0., 1., 0., 0.,
1.]),
array([-59.34624118, -57.58075327, -55.81526536, -
54.04977745, -52.28428954, -50.51880164, -48.75331373, -
46.98782582,
-45.22233791, -43.45685, -41.69136209, -39.92587419,
-38.16038628, -36.39489837, -34.62941046, -32.86392255,
-31.09843464, -29.33294674, -27.56745883, -25.80197092,
-24.03648301, -22.2709951, -20.50550719, -18.74001929,
-16.97453138, -15.20904347, -13.44355556, -11.67806765,
-9.91257974, -8.14709184, -6.38160393, -4.61611602,
-2.85062811, -1.0851402, 0.68034771,
2.44583561, 4.21132352, 5.97681143, 7.74229934,
9.50778725, 11.27327516, 13.03876307,
14.80425097, 16.56973888,
18.33522679, 20.1007147, 21.86620261, 23.63169052,
```

```
25.39717842, 27.16266633, 28.92815424]],
[<matplotlib.patches.Polygon at 0x249e24fe250>])
```



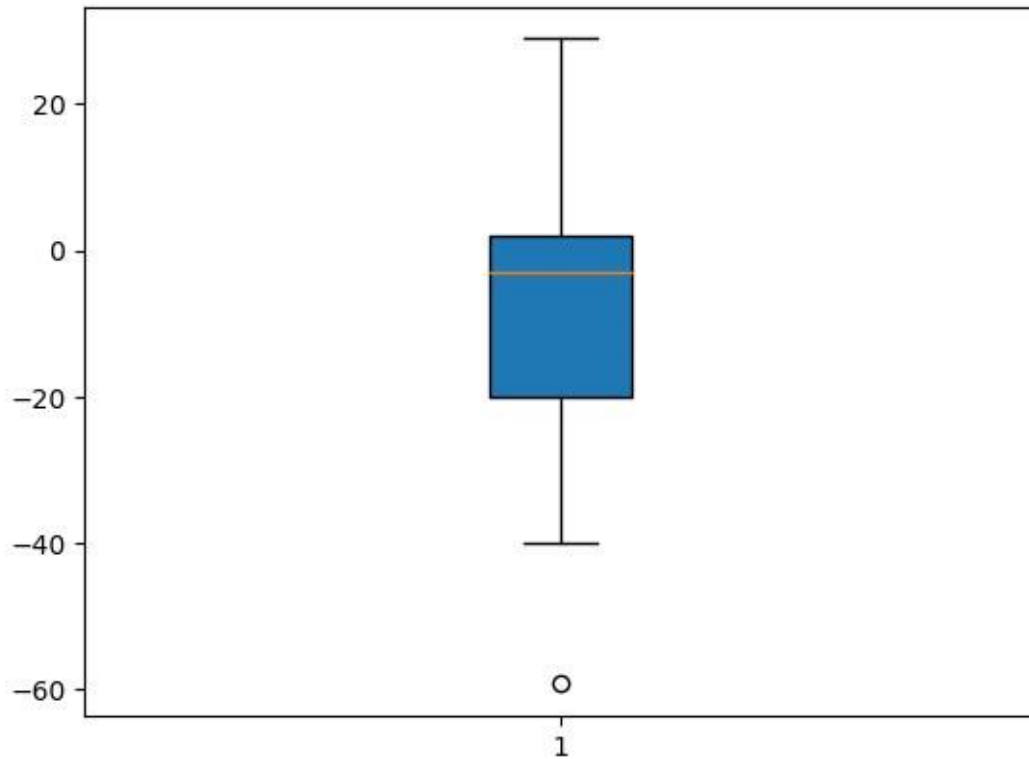
```
[14]: x=[1,2,3,4]
plt.pie(x,explode=(0.1,0,0.1,0))
```

```
[14]: ([<matplotlib.patches.Wedge at
0x249de8e5590>, <matplotlib.patches.Wedge at
0x249e25b7e10>,
<matplotlib.patches.Wedge at 0x249e25c8c50>,
<matplotlib.patches.Wedge at 0x249e25c9d10>],
[Text(1.1412678178182487, 0.3708203985925972, ''),
Text(0.33991867422268784, 1.0461621742897658, ''),
Text(-1.1412678438572794, 0.37082031845269275, ''),
Text(0.3399188211458418, -1.0461621265515308, '')])
```



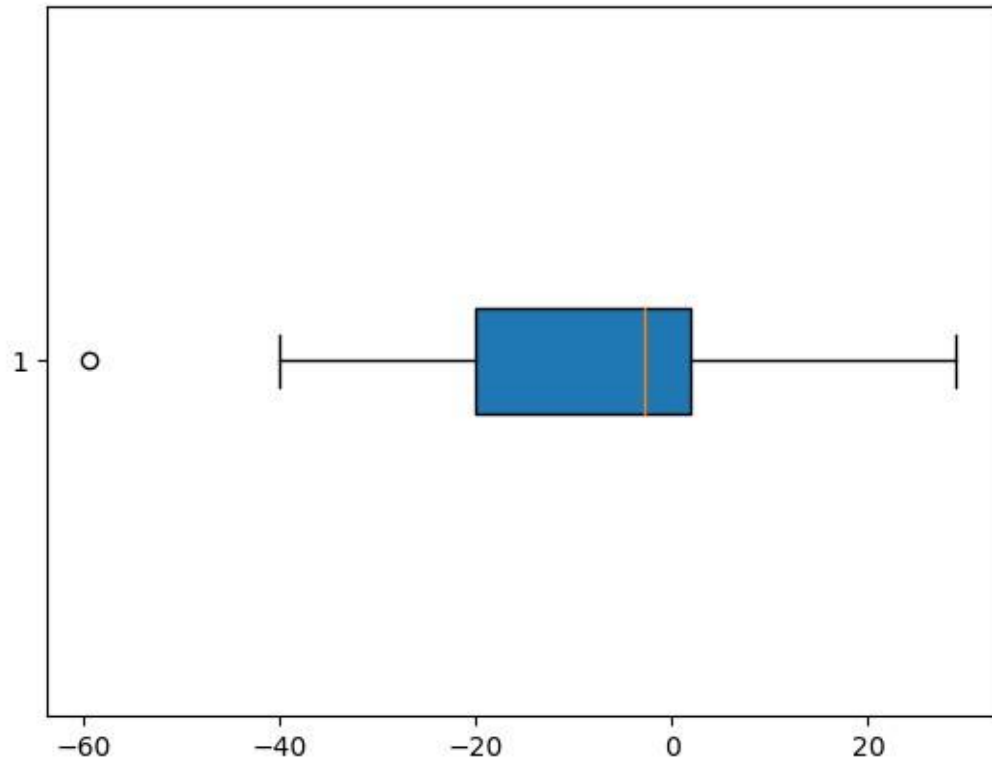
```
[15]: plt.boxplot(y,vert=True,patch_artist=True)
```

```
[15]: {'whiskers': [<matplotlib.lines.Line2D at 0x249de4b04d0>,
<matplotlib.lines.Line2D at 0x249e2602950>],
'caps': [<matplotlib.lines.Line2D at 0x249e26032d0>,
<matplotlib.lines.Line2D at 0x249e2603bd0>],
'boxes': [<matplotlib.patches.PathPatch at
0x249de8e7010>],
'medians': [<matplotlib.lines.Line2D at 0x249e2614450>],
'fliers': [<matplotlib.lines.Line2D at
0x249e24ffc90>], 'means': []}
```



```
[16]: plt.boxplot(y,vert=False,patch_artist=True)
```

```
[16]: {'whiskers': [<matplotlib.lines.Line2D at 0x249e266c190>,
<matplotlib.lines.Line2D at 0x249e266c750>],
'caps': [<matplotlib.lines.Line2D at 0x249e266c050>,
<matplotlib.lines.Line2D at 0x249e266d890>],
'boxes': [<matplotlib.patches.PathPatch at
0x249e2663090>],
'medians': [<matplotlib.lines.Line2D at 0x249e266e150>],
'fliers': [<matplotlib.lines.Line2D at
0x249df9369d0>], 'means': []}
```



Seaborn library for data visualisation `distplot=density+histogram` and it is univariant and it combines histogram with kernel density estimation plot(KDE) plot. `jointplot=`visualise the joint distribution of two variables along with their individual distribution `pairplot=`useful for exploring relationships in multivariate datasets,shows pairwise relationship between variables in dataset `rugplot=`displays individual data points along a single axis `kdeplot=`estimates probability density function of a continuous random variable

```
[17]: import seaborn as sns
```

```
[18]: tips=sns.load_dataset('tips')
tips.head()
```

```
[18]:
```

	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

```
[19]: sns.distplot(tips['total_bill'])
```

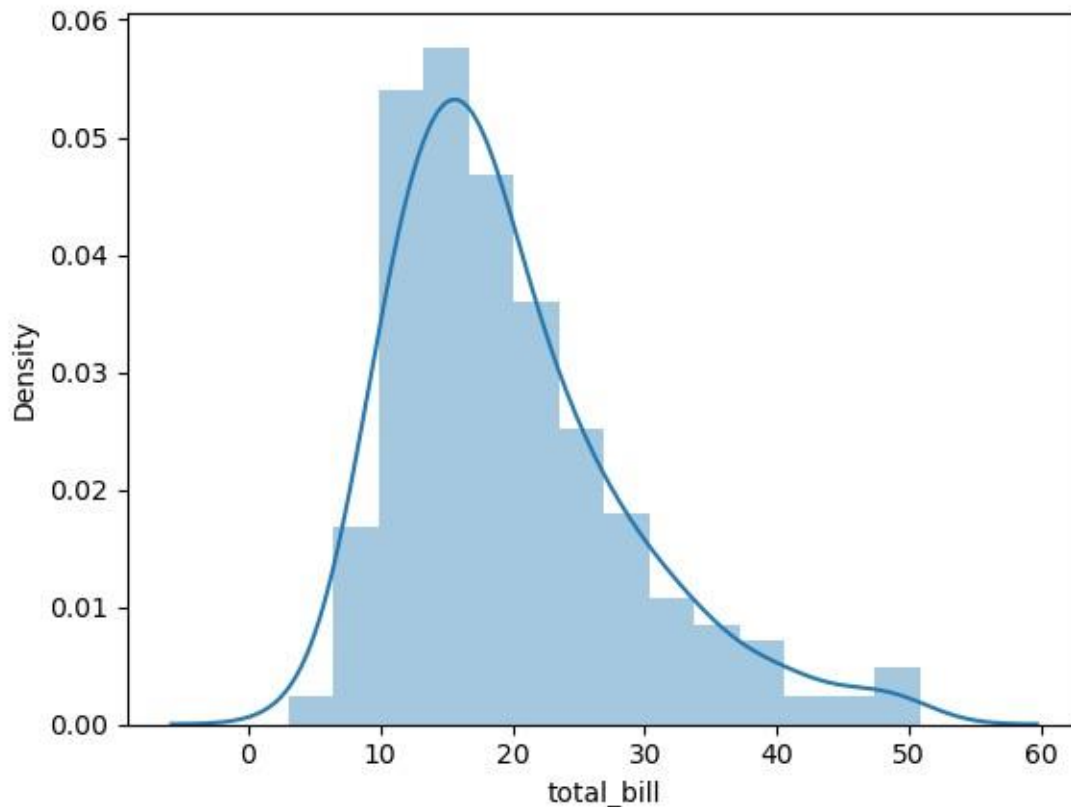
C:\Users\HP\AppData\Local\Temp\ipykernel\_544\4271412032.py:1:  
UserWarning: `distplot` is a deprecated function and will be removed in  
seaborn v0.14.0.

Please adapt your code to use either `displot` (a figure-level function with similar flexibility) or `histplot` (an axes-level function for histograms).

For a guide to updating your code to use the new functions, please see <https://gist.github.com/mwaskom/de44147ed2974457ad6372750bbe5751>

```
sns.distplot(tips['total_bill'])
```

```
[19]: <Axes: xlabel='total_bill', ylabel='Density'>
```

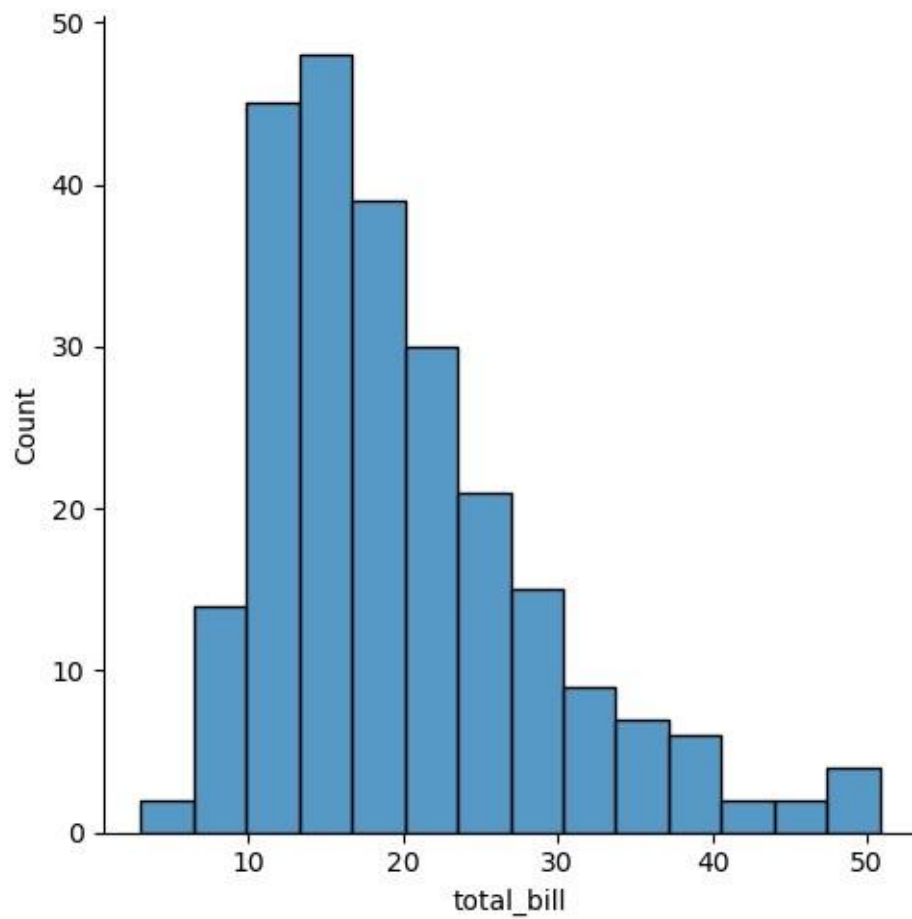


```
[20]: sns.displot(tips['total_bill'])
```

```
C:\Users\HP\anaconda3\Lib\site-packages\seaborn\axisgrid.py:118:
UserWarning:
The figure layout has changed to tight
  self._figure.tight_layout(*args, **kwargs)
```

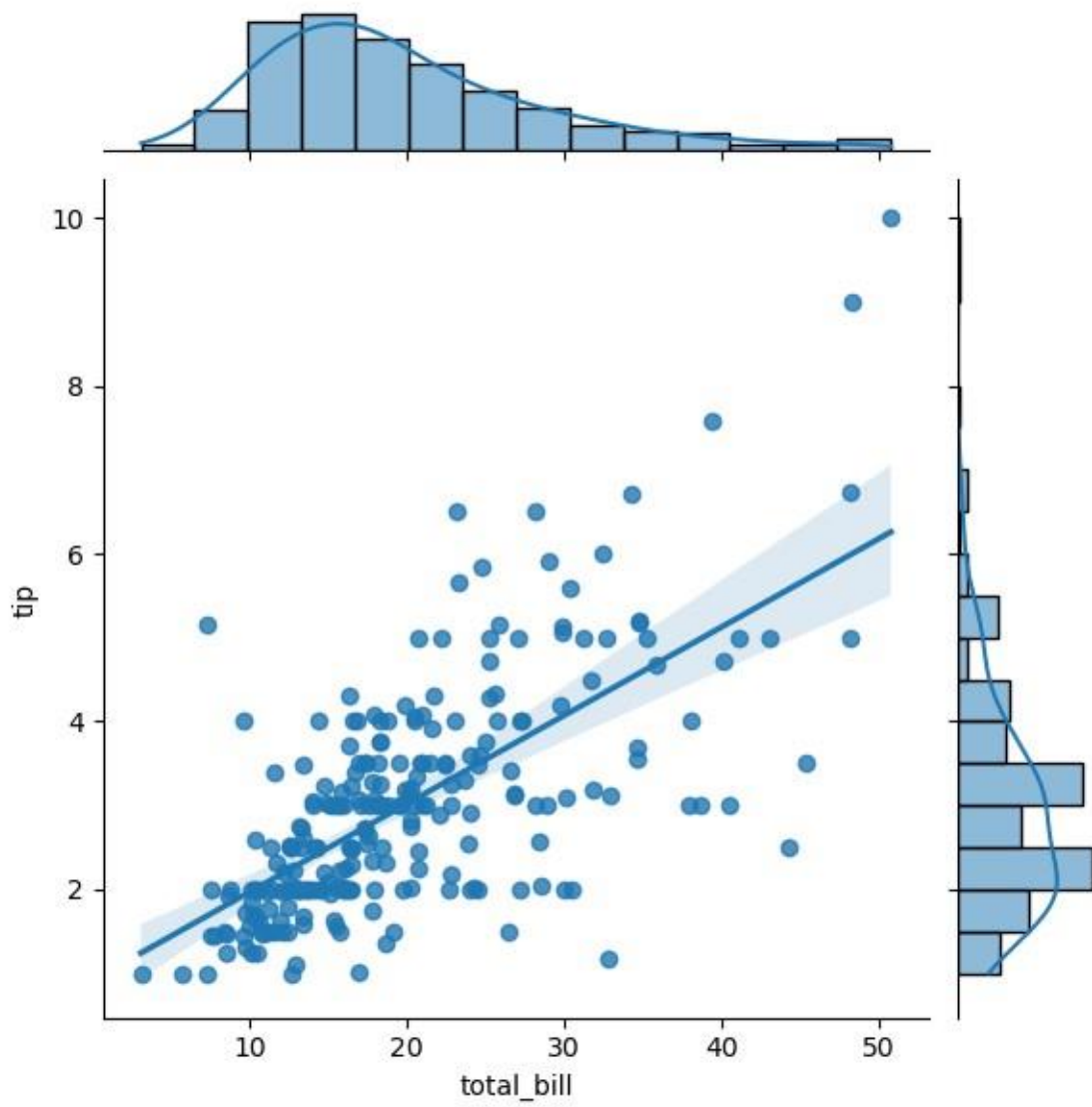
```
[20]: <seaborn.axisgrid.FacetGrid at 0x249e3e4f750>
```





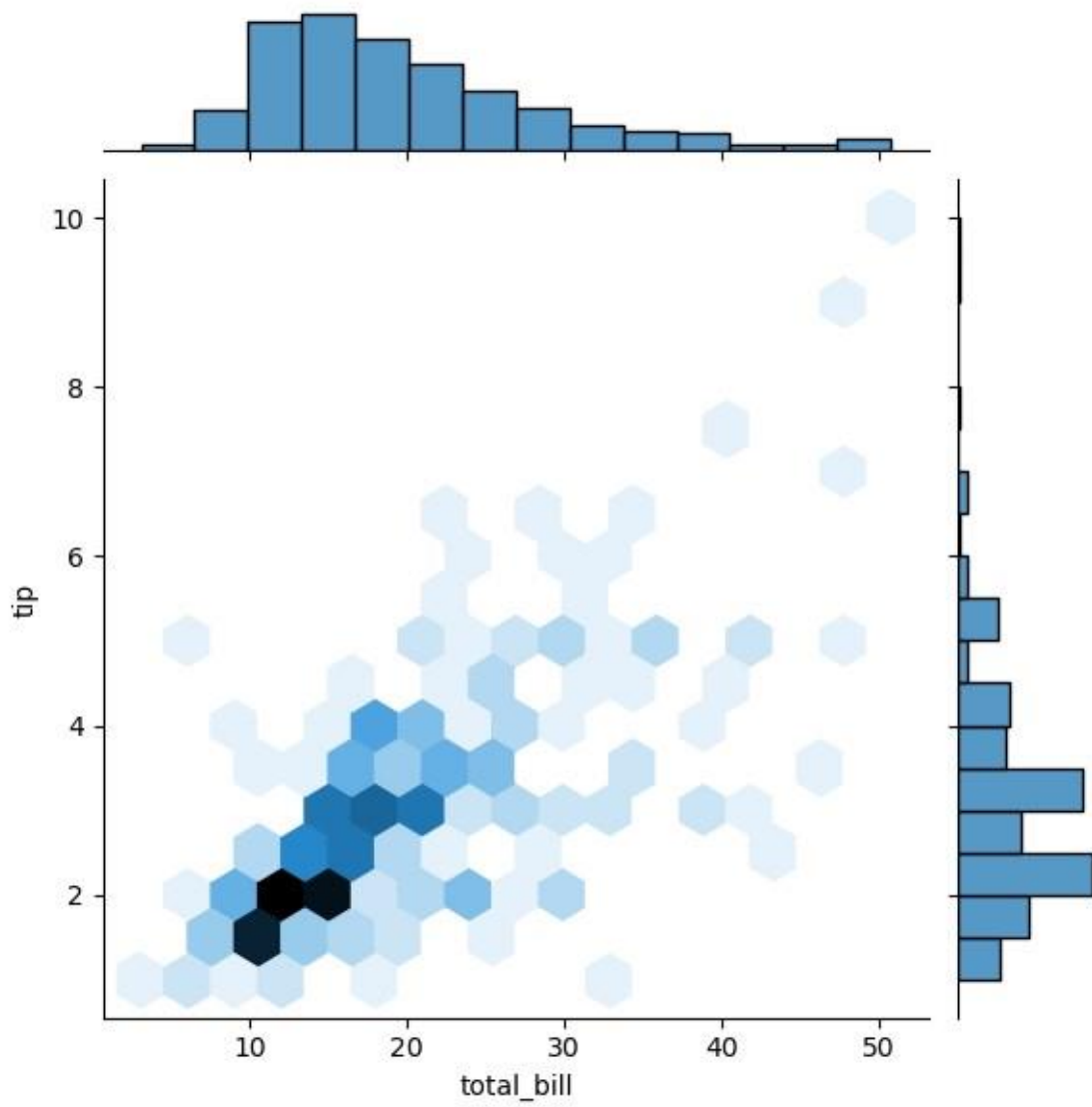
```
[21]: sns.jointplot(x='total_bill',y='tip',data=tips,kind='reg')
```

```
[21]: <seaborn.axisgrid.JointGrid at 0x249e3e6a690>
```



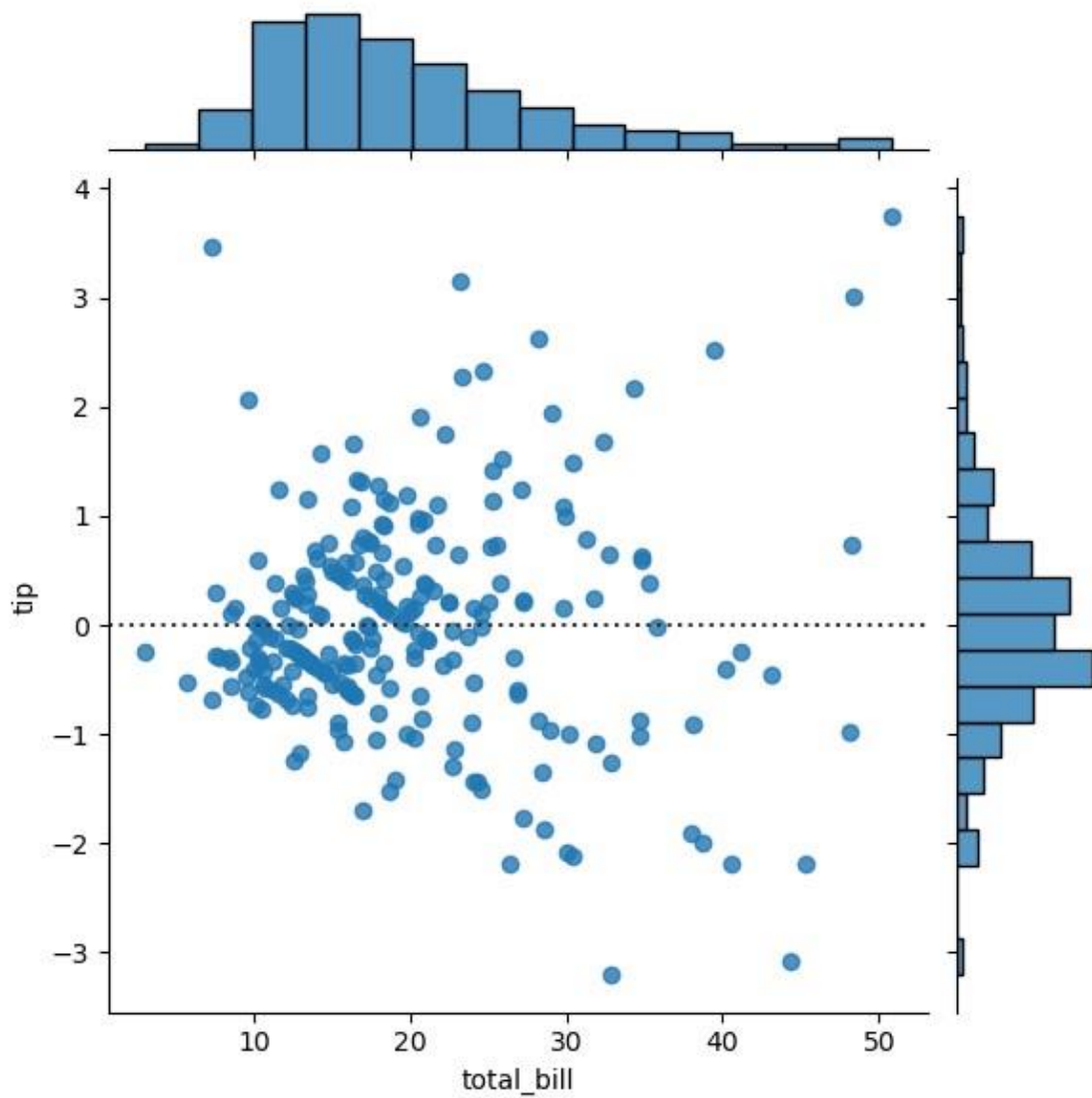
```
[22]: sns.jointplot(x='total_bill',y='tip',data=tips,kind='hex')
```

```
[22]: <seaborn.axisgrid.JointGrid at 0x249e2682fd0>
```



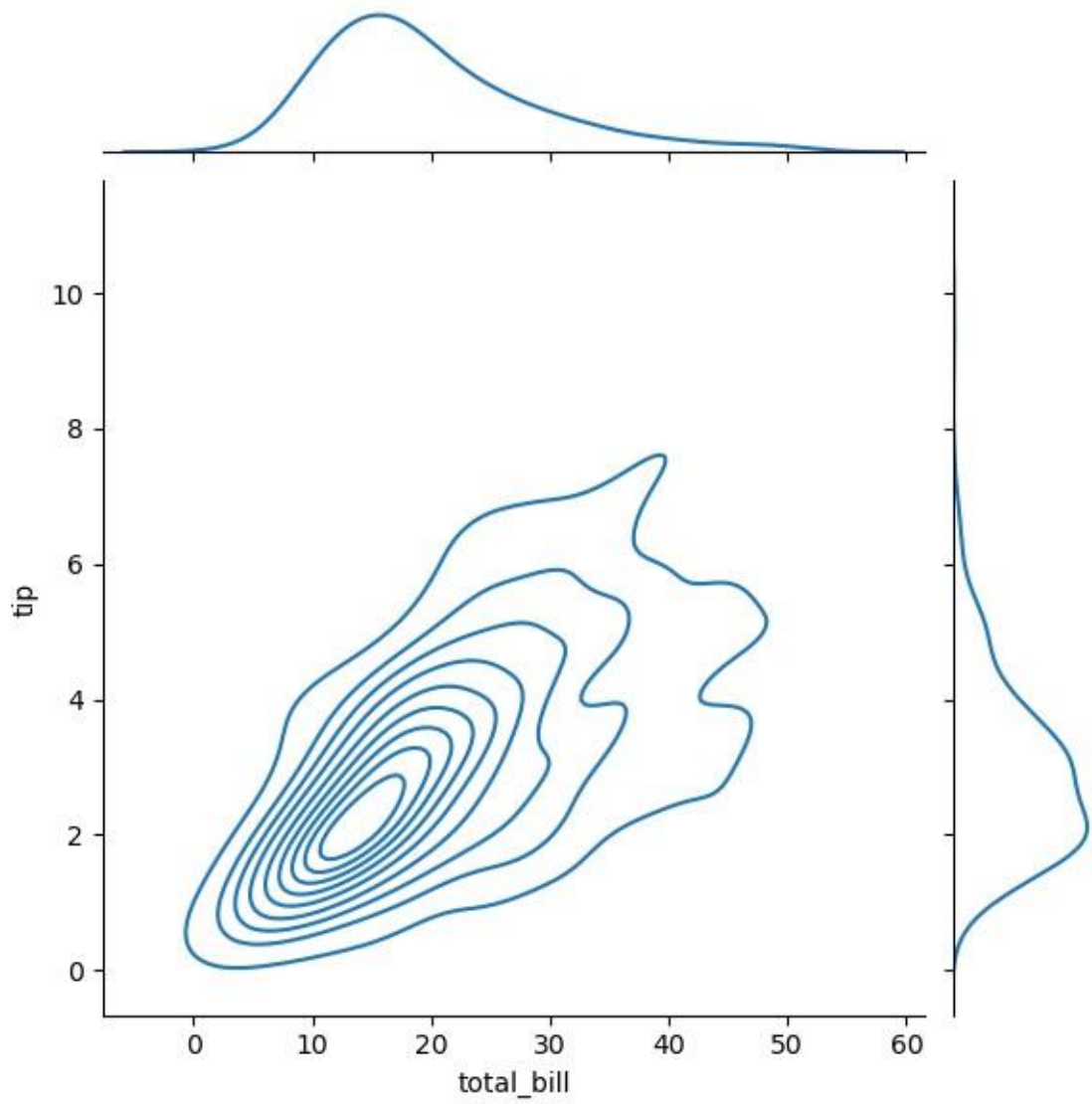
```
[23]: sns.jointplot(x='total_bill',y='tip',data=tips,kind='resid')
```

```
[23]: <seaborn.axisgrid.JointGrid at 0x249e5c94e10>
```



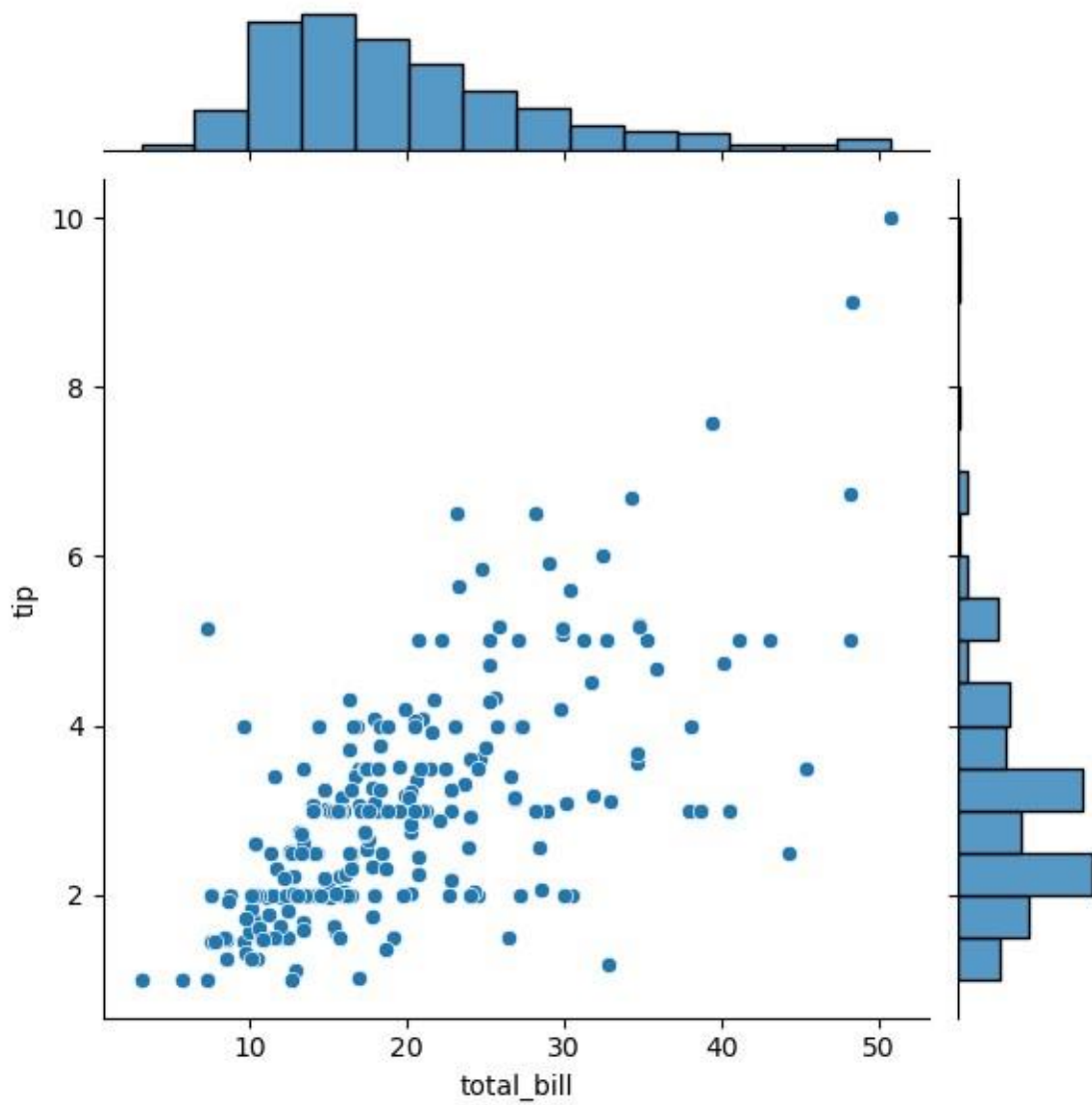
```
[24]: sns.jointplot(x='total_bill',y='tip',data=tips,kind='kde')
```

```
[24]: <seaborn.axisgrid.JointGrid at 0x249e39ff310>
```



```
[25]: sns.jointplot(x='total_bill',y='tip',data=tips,kind='scatter')
```

```
[25]: <seaborn.axisgrid.JointGrid at 0x249e655b410>
```

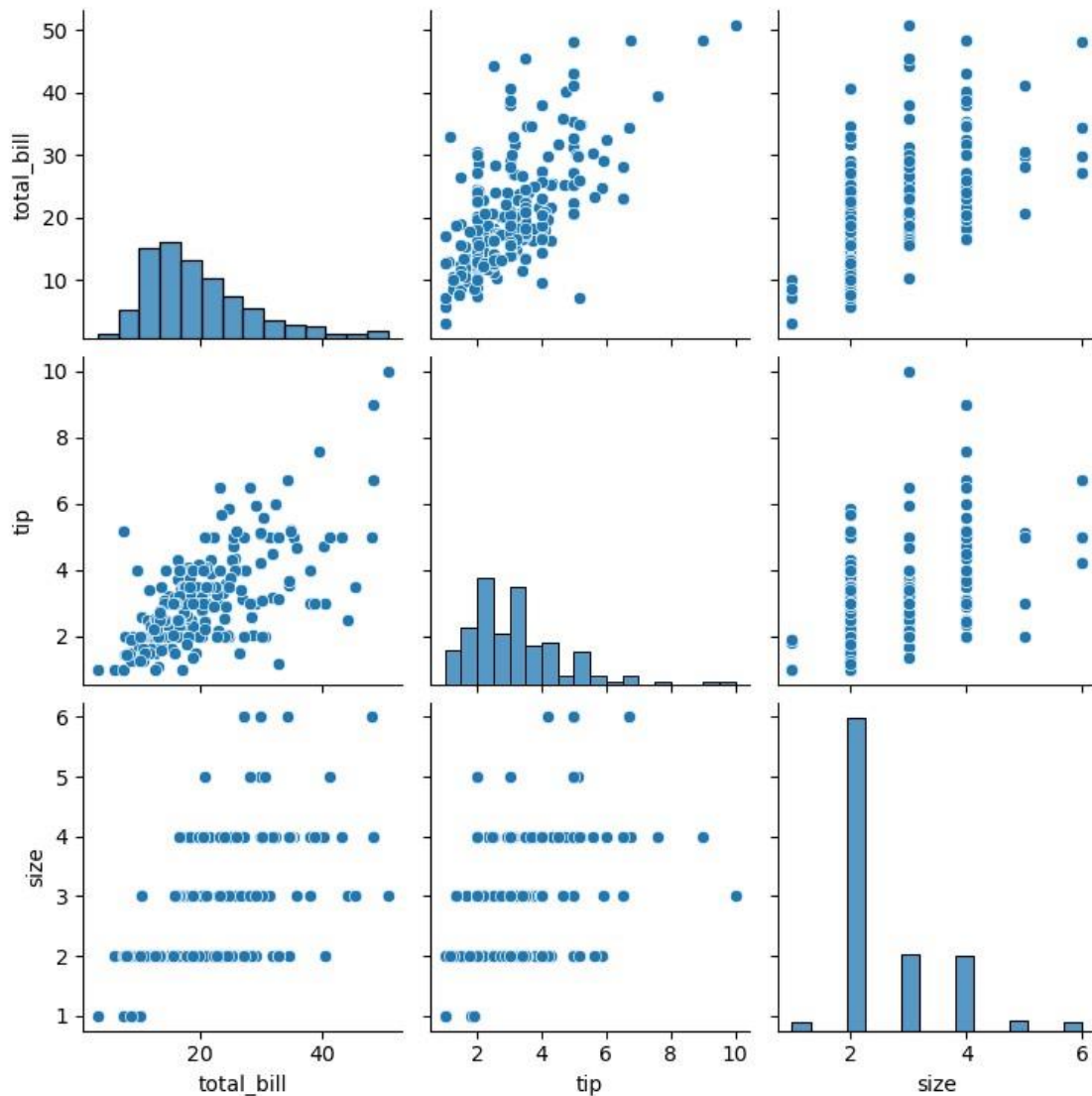


```
[26]: sns.pairplot(tips)
```

```
C:\Users\HP\anaconda3\Lib\site-packages\seaborn\axisgrid.py:118: UserWarning:
```

```
The figure layout has changed to tight
self._figure.tight_layout(*args, **kwargs)
```

```
[26]: <seaborn.axisgrid.PairGrid at 0x249e5f3c950>
```

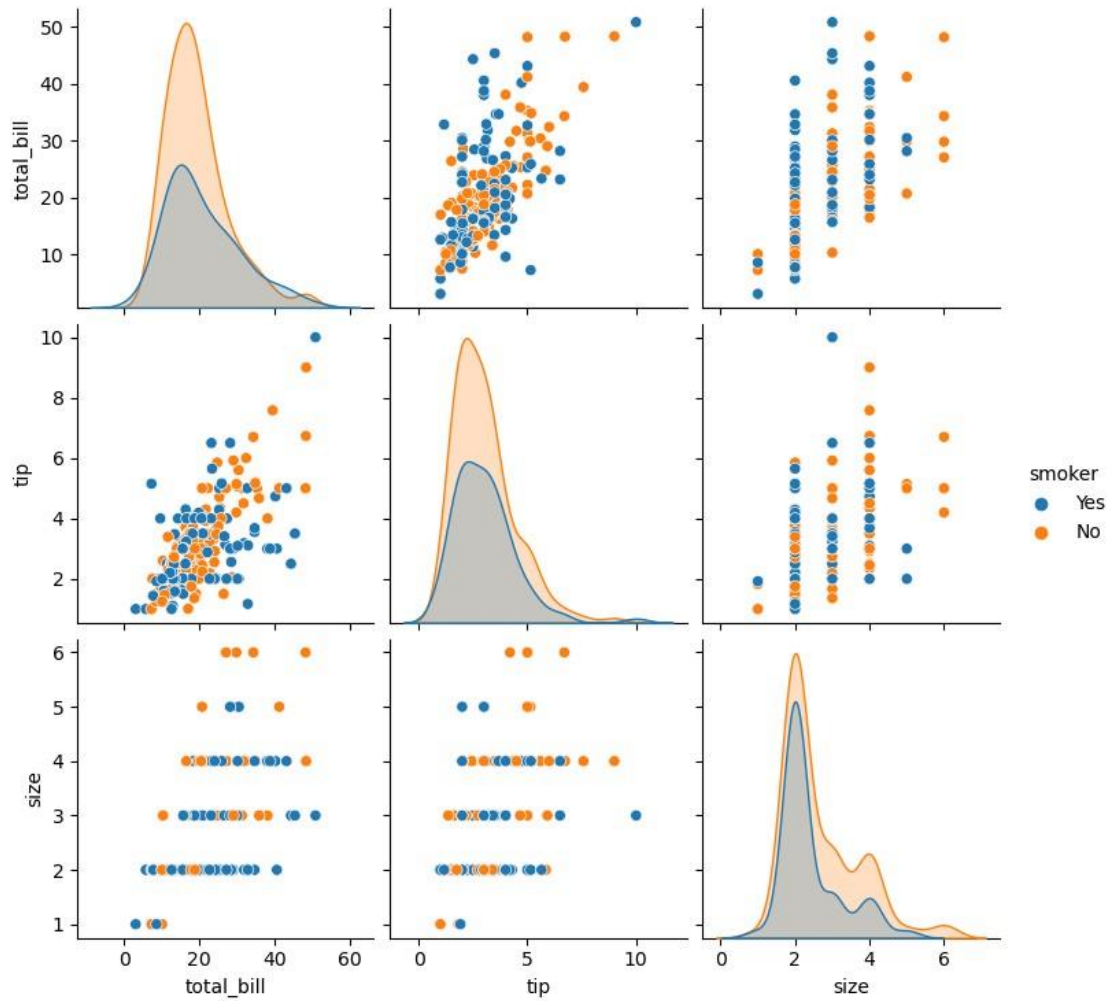


```
[27]: sns.pairplot(tips, hue='smoker')
```

```
C:\Users\HP\anaconda3\Lib\site-packages\seaborn\axisgrid.py:118:
UserWarning:
```

```
The figure layout has changed to tight
self._figure.tight_layout(*args, **kwargs)
```

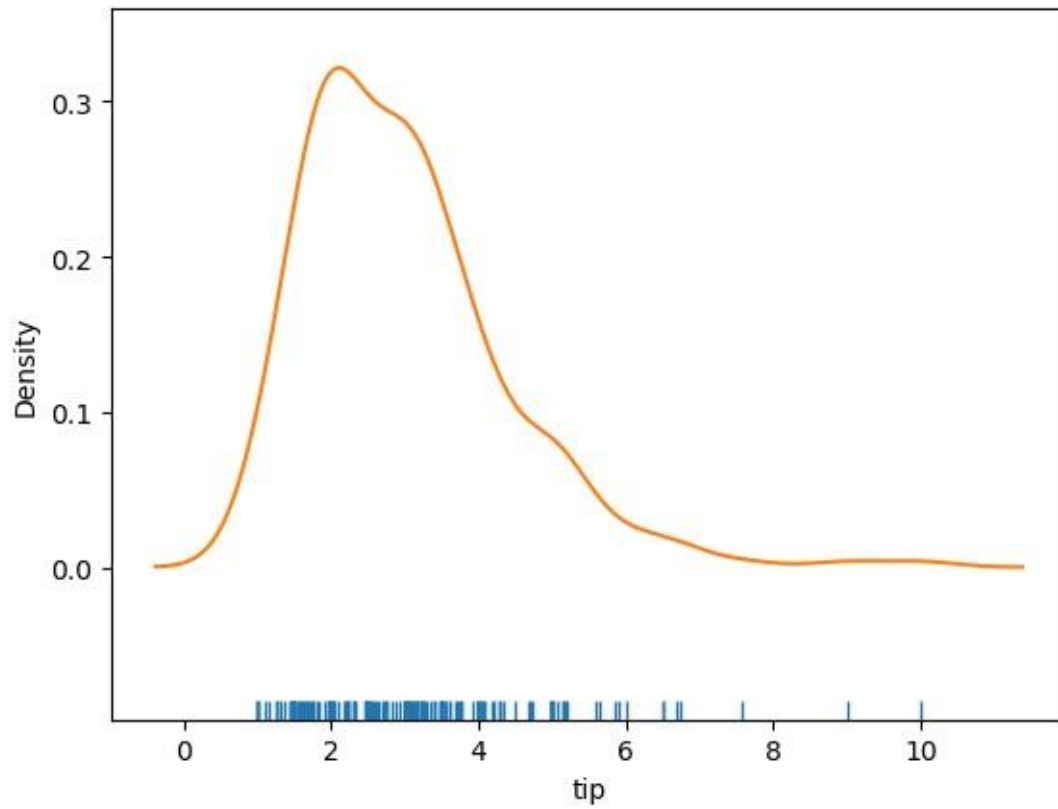
```
[27]: <seaborn.axisgrid.PairGrid at 0x249e7364610>
```



```
[28]: sns.rugplot(tips['tip'])
      sns.kdeplot(tips['tip'])
```

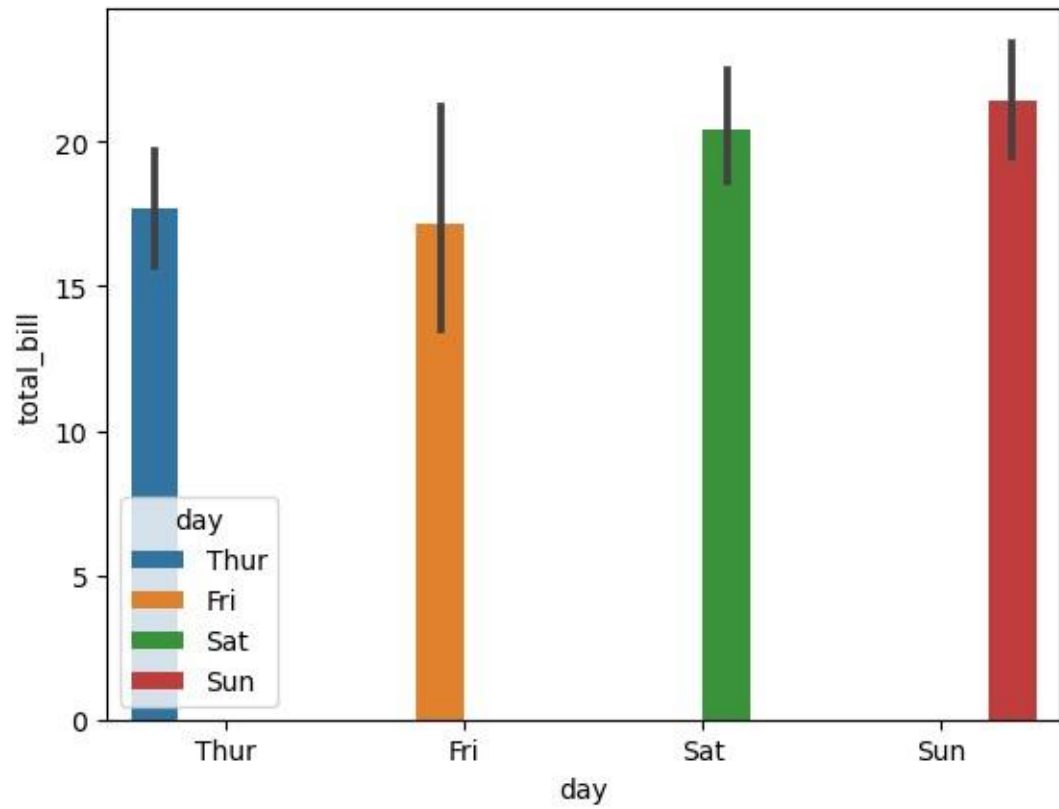
```
[28]: <Axes: xlabel='tip', ylabel='Density'>
```





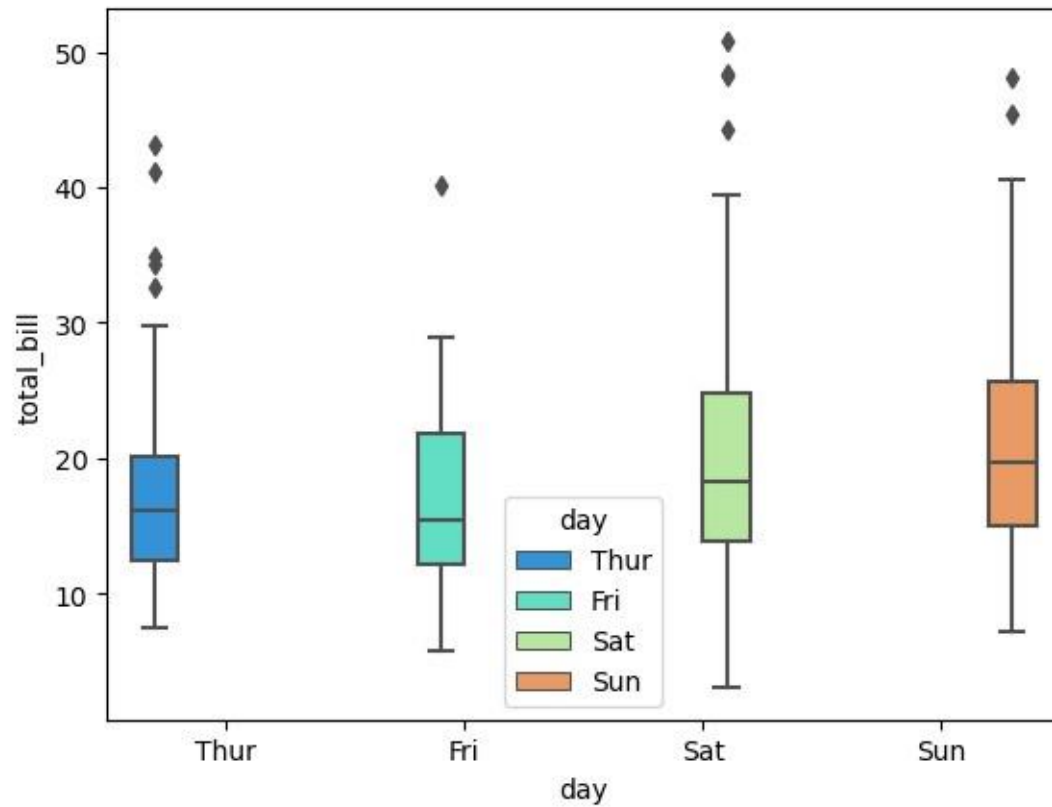
```
[29]: sns.barplot(x='day',y='total_bill',data=tips,hue='day')
```

```
[29]: <Axes: xlabel='day', ylabel='total_bill'>
```



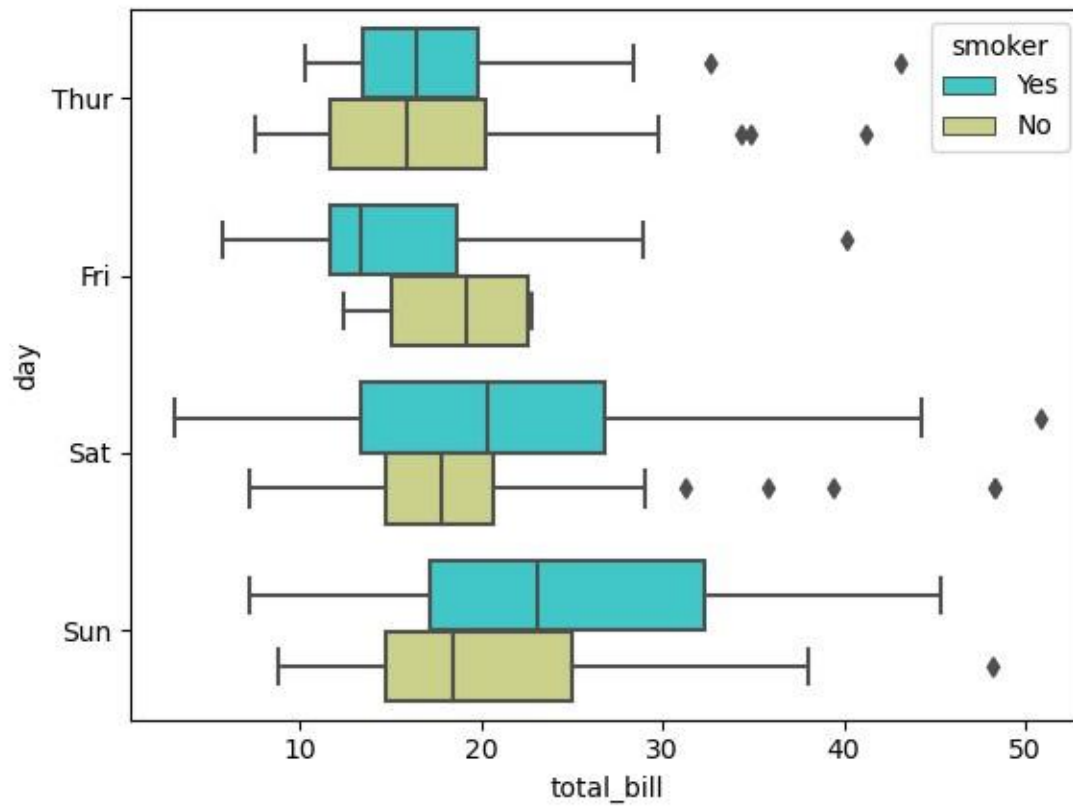
```
[30]: sns.boxplot(x='day',y='total_bill',data=tips,palette='rainbow',hue='day')
```

```
[30]: <Axes: xlabel='day', ylabel='total_bill'>
```



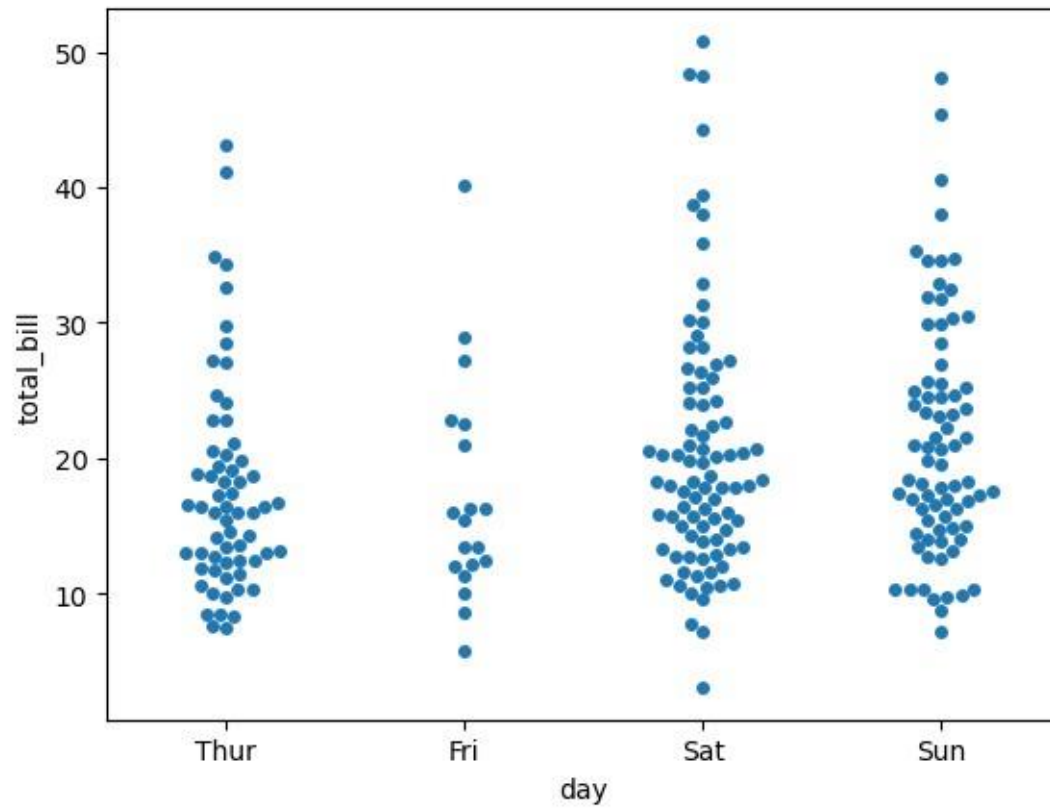
```
[31]: sns.  
      boxplot(x='total_bill',y='day',data=tips,palette='rainbow',hue='smoker',orient  
              ='h')
```

```
[31]: <Axes: xlabel='total_bill', ylabel='day'>
```



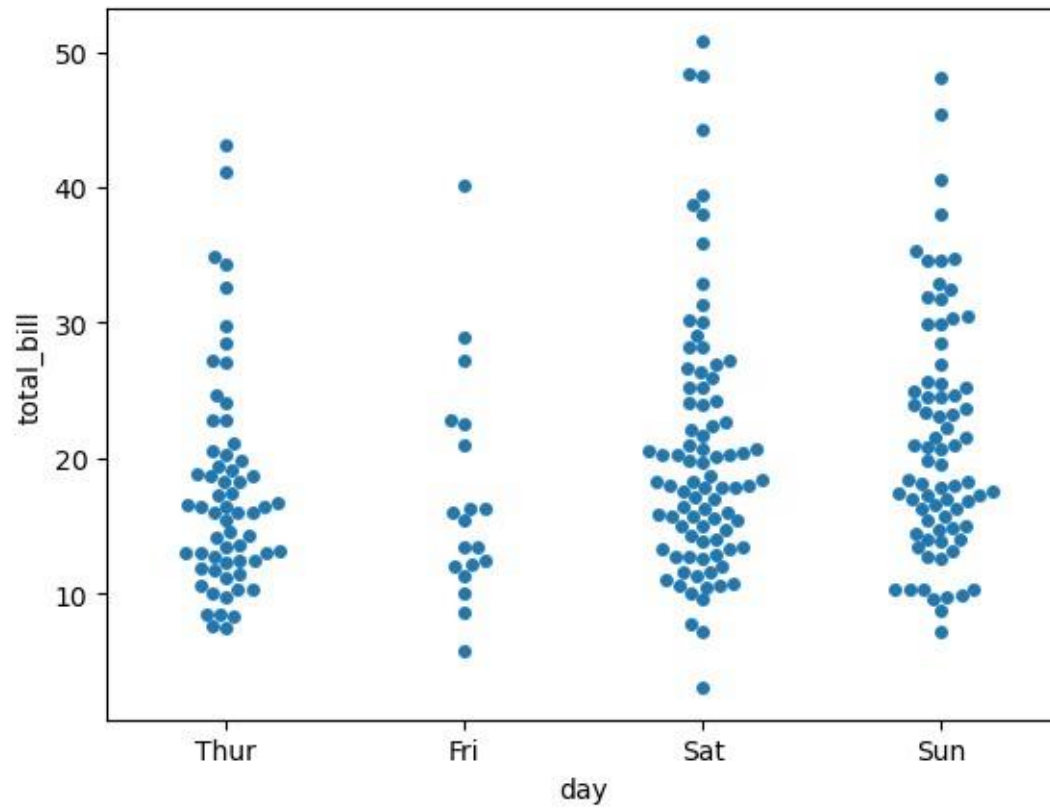
```
[32]: sns.swarmplot(x='day',y='total_bill',data=tips)
```

```
[32]: <Axes: xlabel='day', ylabel='total_bill'>
```



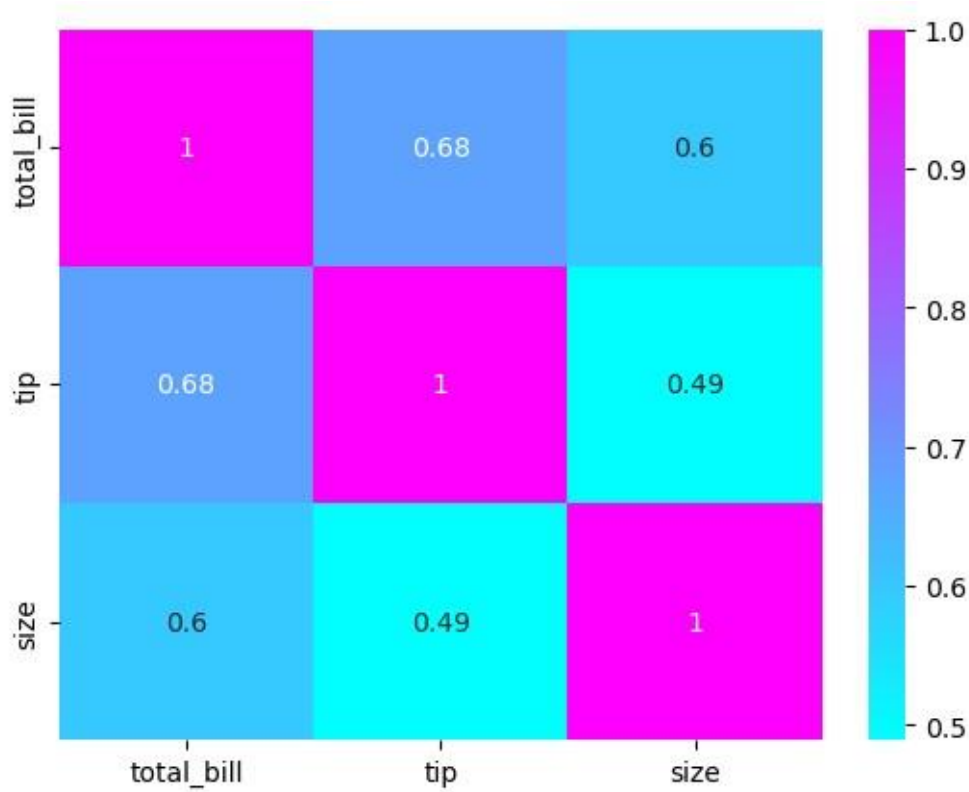
```
[33]: sns.swarmplot(x='day',y='total_bill',data=tips,dodge=True)
```

```
[33]: <Axes: xlabel='day', ylabel='total_bill'>
```



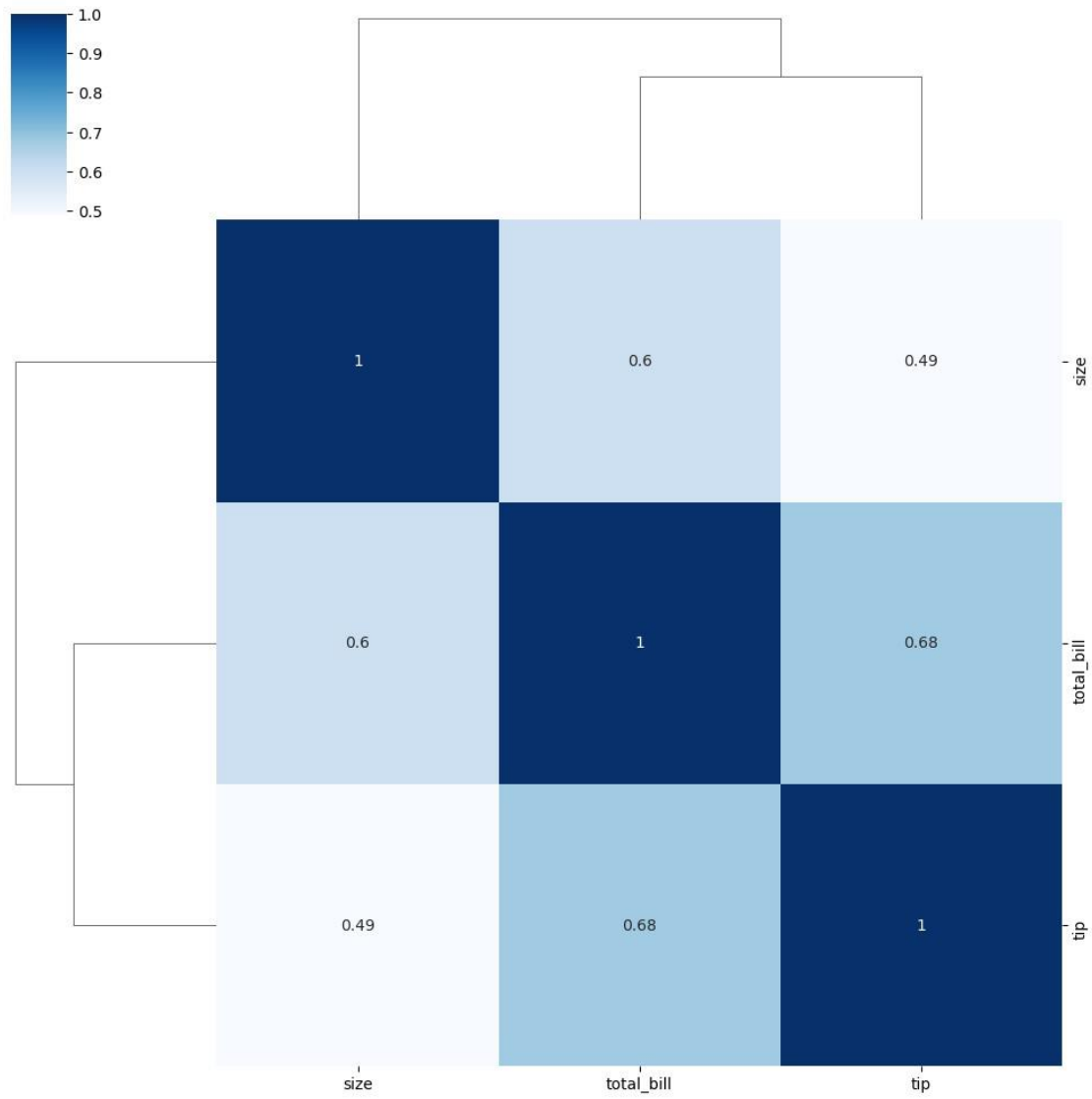
```
[34]: sns.heatmap(tips.corr(numeric_only=True),annot=True,cmap='cool')
```

```
[34]: <Axes: >
```



```
[35]: sns.clustermap(tips.corr(numeric_only=True), annot=True, cmap="Blues")
```

```
[35]: <seaborn.matrix.ClusterGrid at 0x249e887c790>
```



[ ]: