In [110]:
```
df
```

Out[110]:

	Name	OverallGrade	Obedient	ResearchScore	ProjectScore
0	Henry	1	1	95	185
1	John	3	0	90	151
2	David	5	0	15	122
3	Holmes	2	1	80	171
4	Marvin	4	0	25	135
5	Simon	1	1	97	179
6	Robert	2	1	65	159
7	Trent	3	1	80	138

Matplotlib

In [1]:
```
import matplotlib.pyplot as plt
```

In [2]:
```
plt.style.use('classic')
# http://tonysyu.github.io/raw_content/matplotlib-style-gallery/gallery.html
```

In [3]:
```
import numpy as np
```

In [5]:
```
np.linspace(0,10,5)
```

Out[5]:
```
array([ 0. ,  2.5,  5. ,  7.5, 10. ])
```

```
In [6]: np.linspace(0,10,10)
```

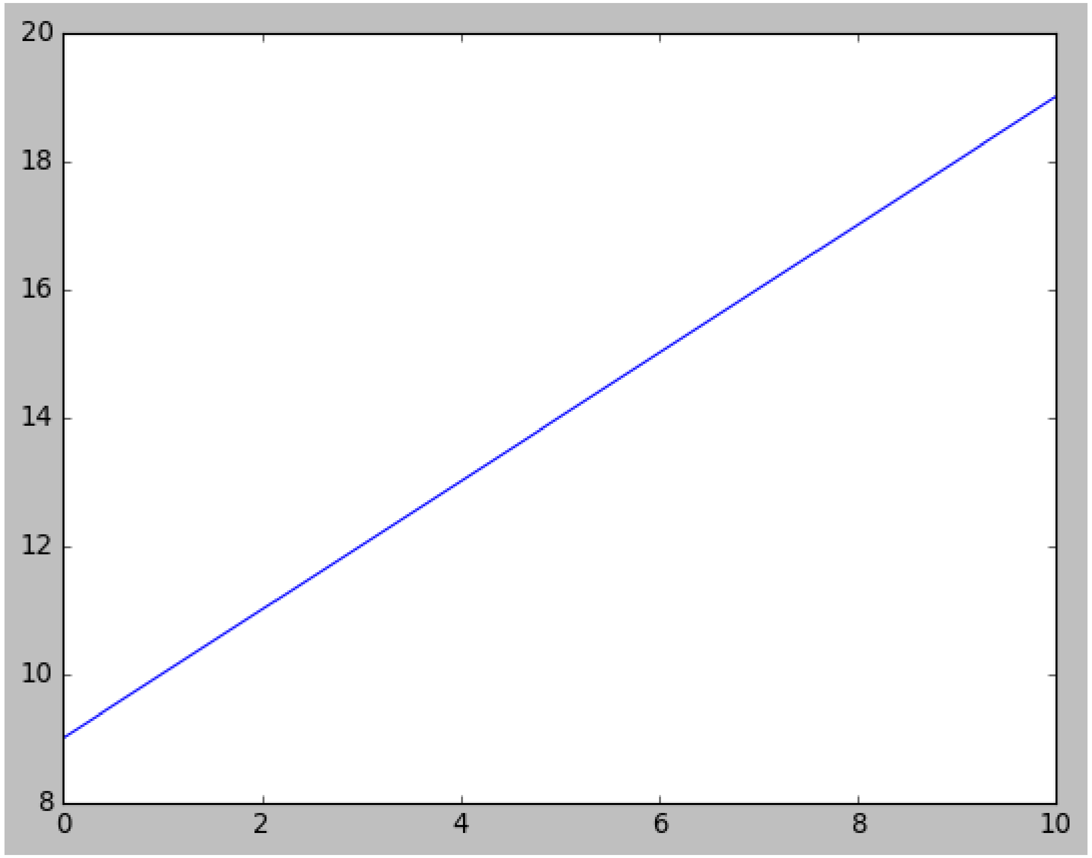
```
Out[6]: array([ 0.        ,  1.11111111,  2.22222222,  3.33333333,  4.44444444,
                5.55555556,  6.66666667,  7.77777778,  8.88888889, 10.        ])
```

```
In [7]: x = np.linspace(0,10,100)
```

```
In [9]: x
```

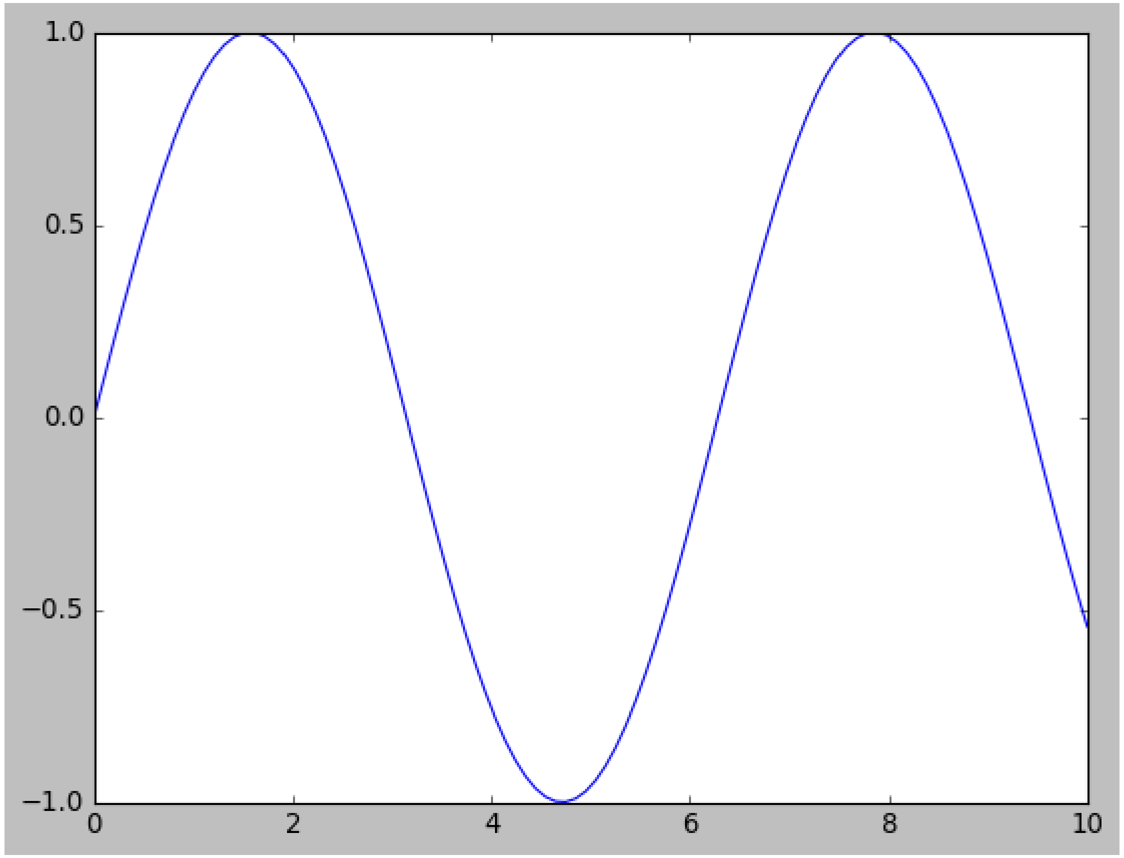
```
Out[9]: array([ 0.        ,  0.1010101 ,  0.2020202 ,  0.3030303 ,  0.4040404 ,
                0.50505051,  0.60606061,  0.70707071,  0.80808081,  0.90909091,
                1.01010101,  1.11111111,  1.21212121,  1.31313131,  1.41414141,
                1.51515152,  1.61616162,  1.71717172,  1.81818182,  1.91919192,
                2.02020202,  2.12121212,  2.22222222,  2.32323232,  2.42424242,
                2.52525253,  2.62626263,  2.72727273,  2.82828283,  2.92929293,
                3.03030303,  3.13131313,  3.23232323,  3.33333333,  3.43434343,
                3.53535354,  3.63636364,  3.73737374,  3.83838384,  3.93939394,
                4.04040404,  4.14141414,  4.24242424,  4.34343434,  4.44444444,
                4.54545455,  4.64646465,  4.74747475,  4.84848485,  4.94949495,
                5.05050505,  5.15151515,  5.25252525,  5.35353535,  5.45454545,
                5.55555556,  5.65656566,  5.75757576,  5.85858586,  5.95959596,
                6.06060606,  6.16161616,  6.26262626,  6.36363636,  6.46464646,
                6.56565657,  6.66666667,  6.76767677,  6.86868687,  6.96969697,
                7.07070707,  7.17171717,  7.27272727,  7.37373737,  7.47474747,
                7.57575758,  7.67676768,  7.77777778,  7.87878788,  7.97979798,
                8.08080808,  8.18181818,  8.28282828,  8.38383838,  8.48484848,
                8.58585859,  8.68686869,  8.78787879,  8.88888889,  8.98989899,
                9.09090909,  9.19191919,  9.29292929,  9.39393939,  9.49494949,
                9.5959596 ,  9.6969697 ,  9.7979798 ,  9.8989899 , 10.        ])
```

In [10]:
```
plt.plot(x, x+9);
```

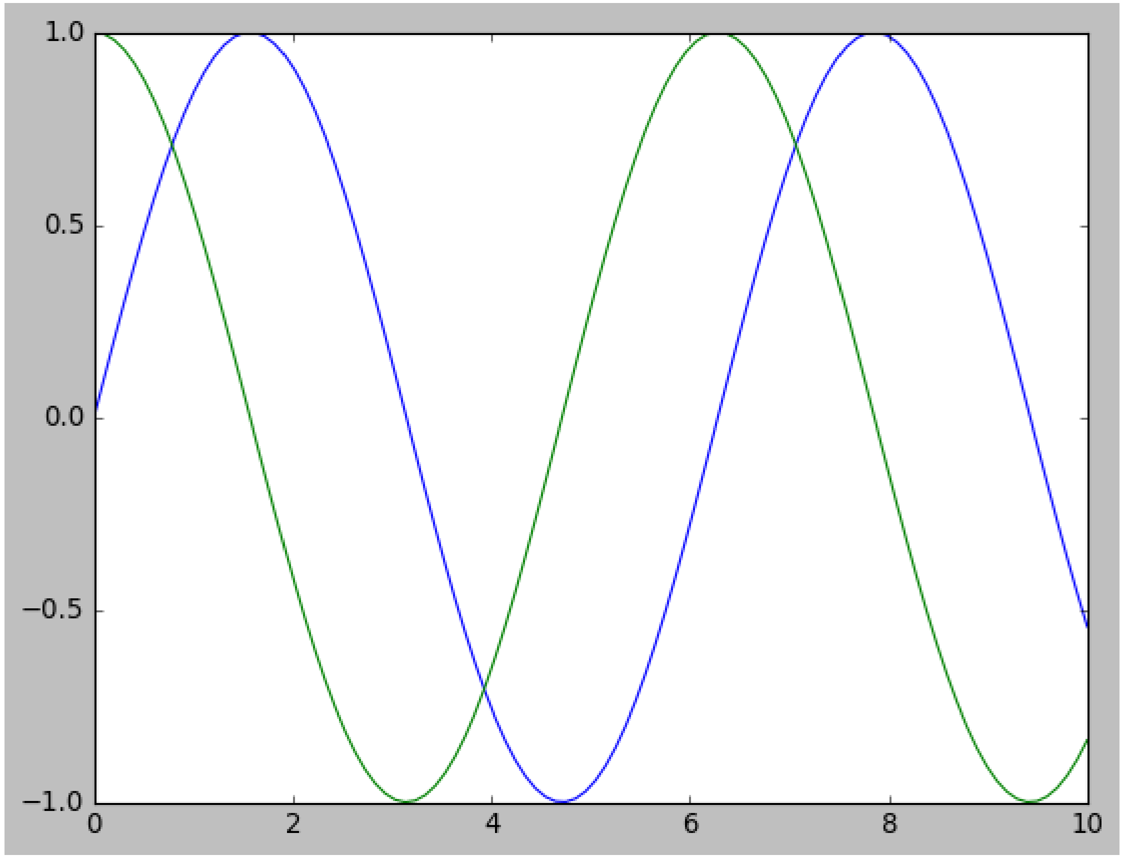


In [11]:

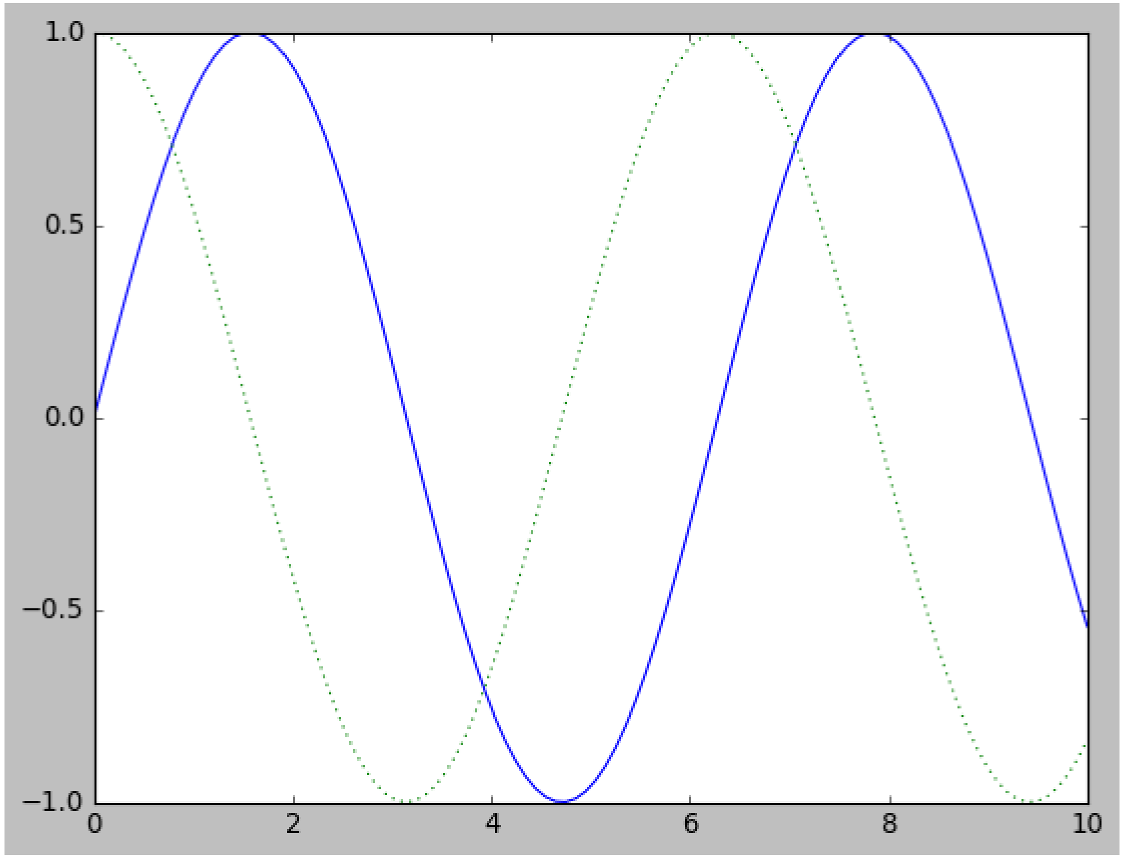
```
plt.plot(x, np.sin(x));
```



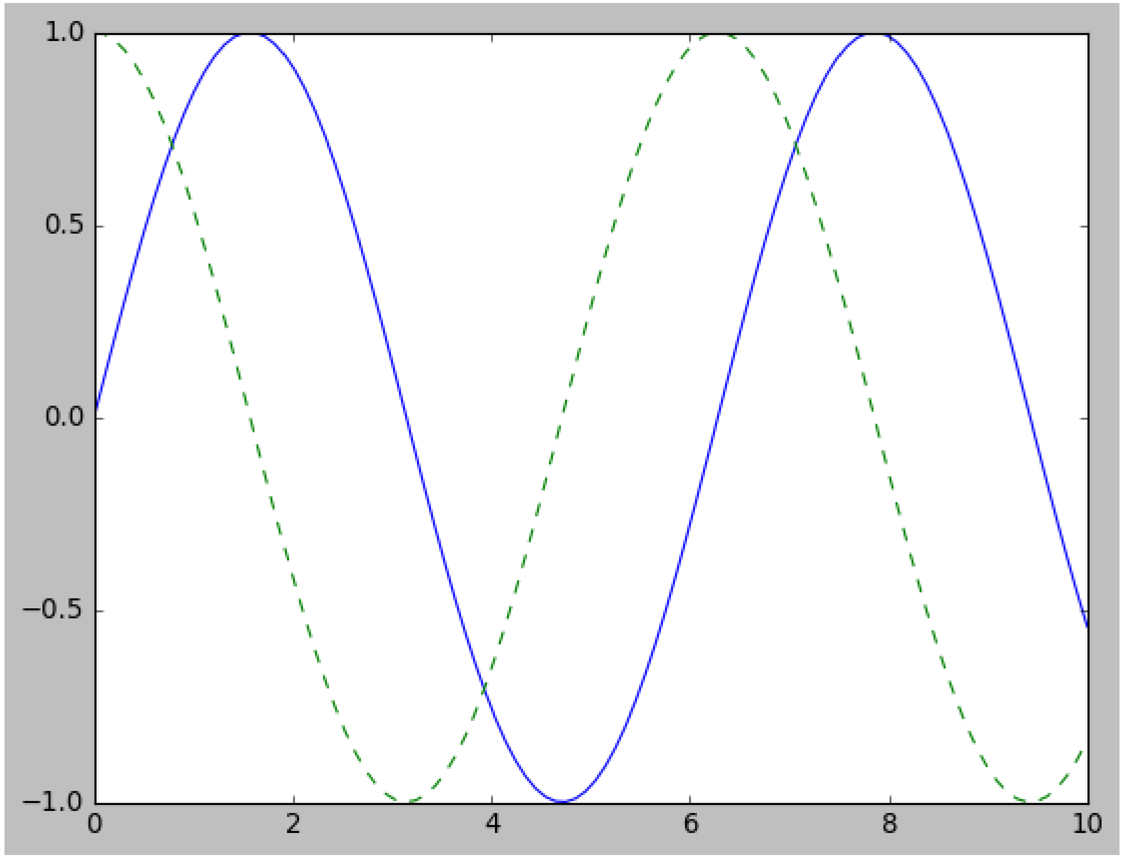
In [12]:
```
plt.plot(x, np.sin(x))
plt.plot(x, np.cos(x));
```



In [15]:
```
plt.plot(x, np.sin(x),'-')
plt.plot(x, np.cos(x),':');
```



In [16]:
```
plt.plot(x, np.sin(x),'-')
plt.plot(x, np.cos(x),'--');
```



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

In [19]:

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

Out[19]:

	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

In [20]:

```
iris.shape
```

Out[20]:

```
(150, 5)
```

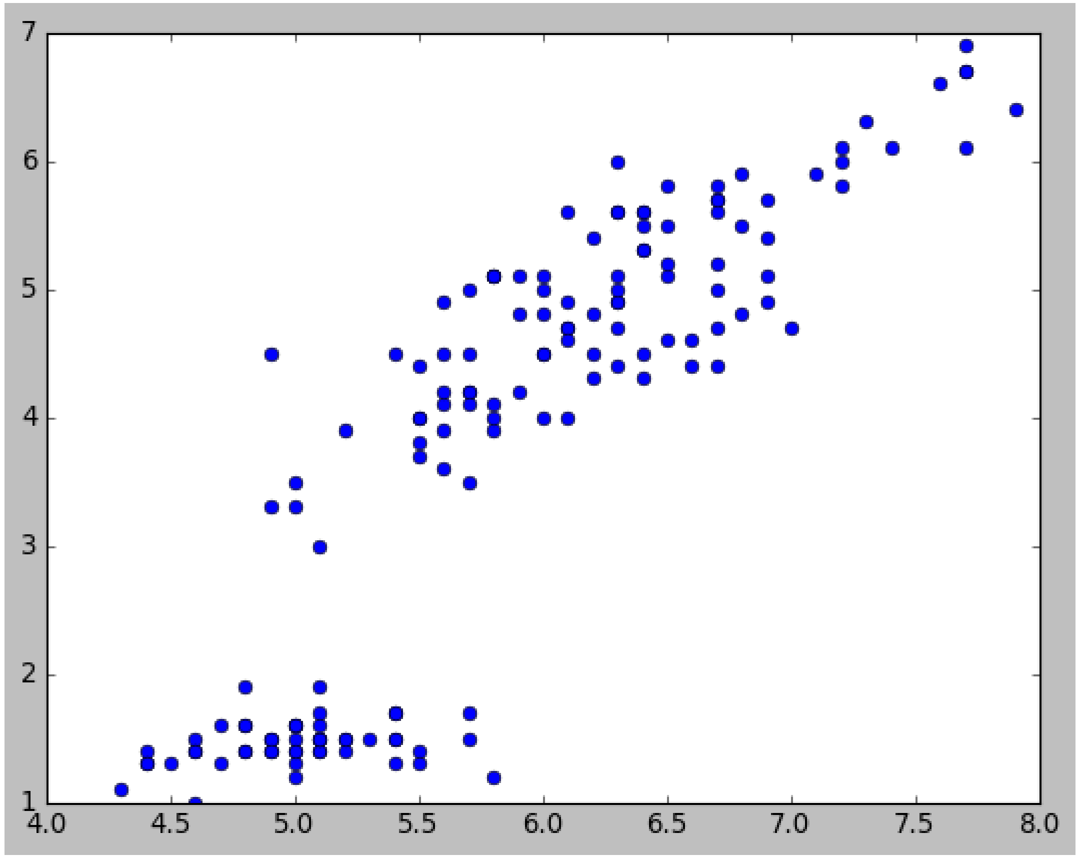
In [21]:

```
iris.species.value_counts()
```

Out[21]:

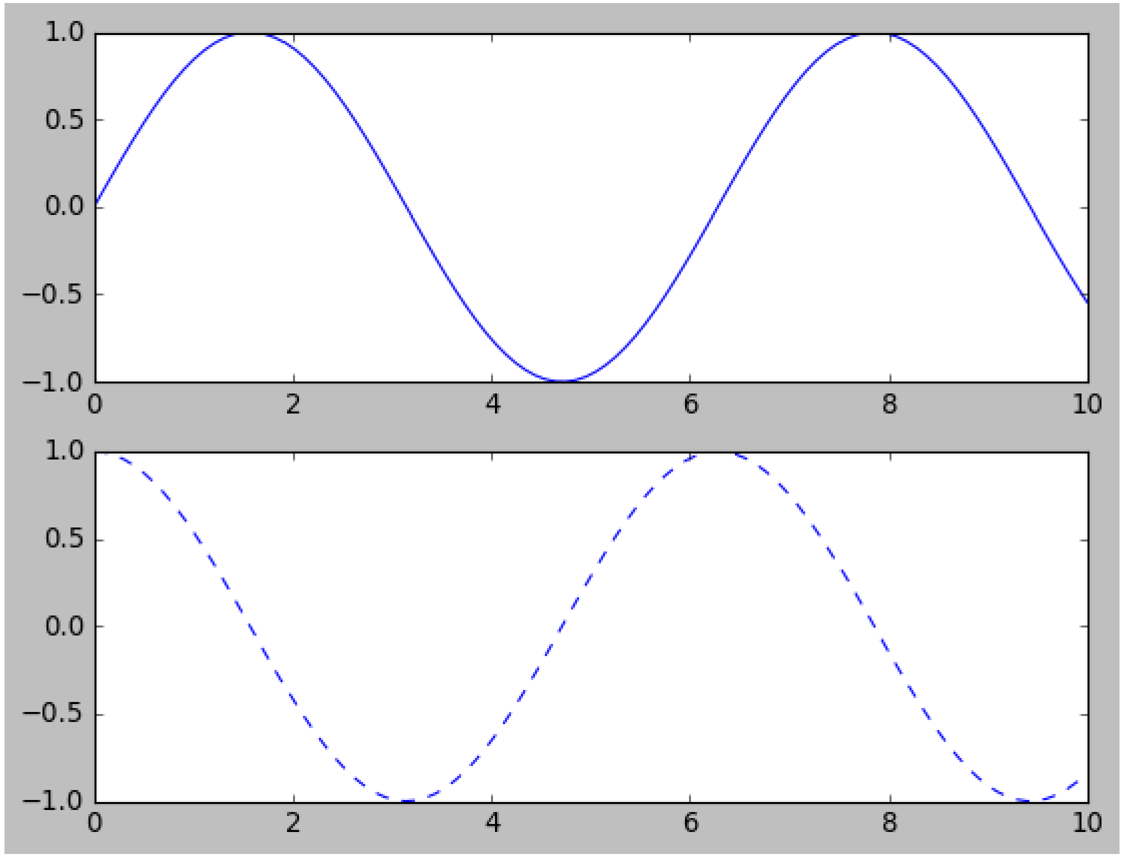
```
setosa        50
versicolor    50
virginica     50
Name: species, dtype: int64
```

```
In [25]: plt.plot(iris['sepal_length'], iris['petal_length'],"o");
```

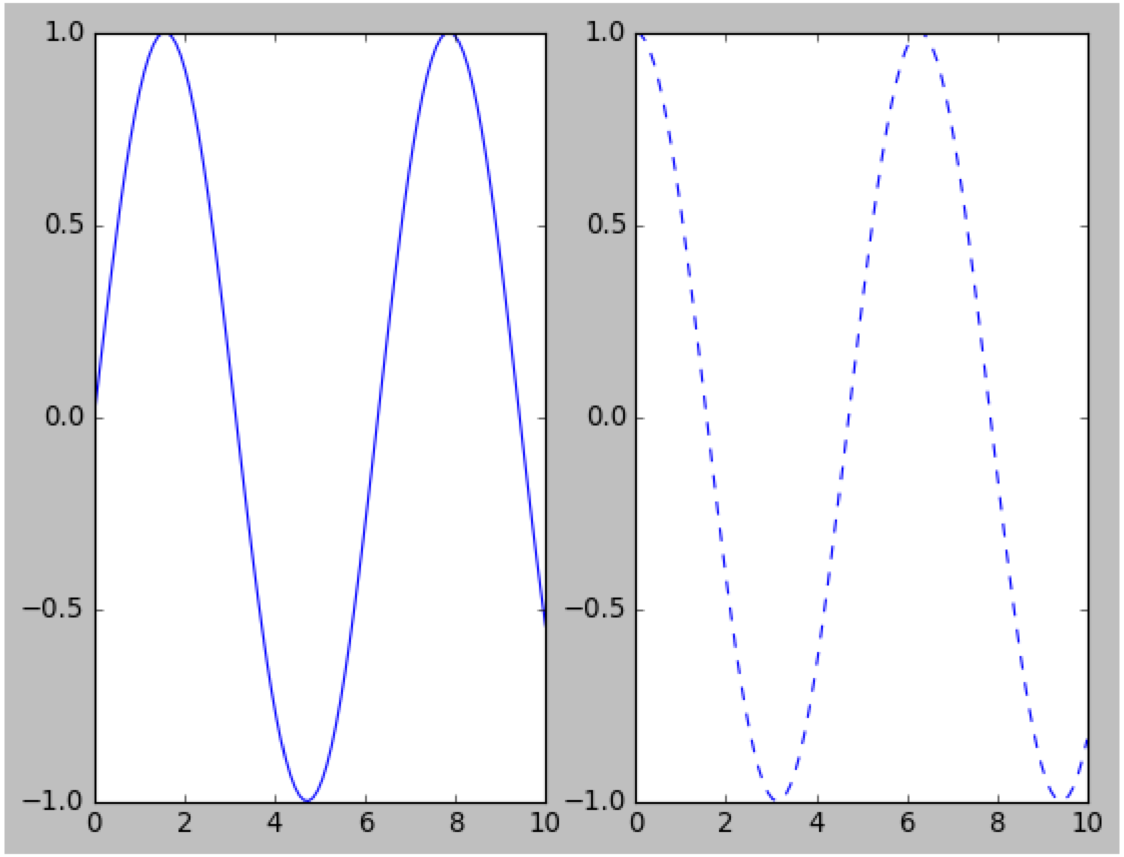


In [27]:

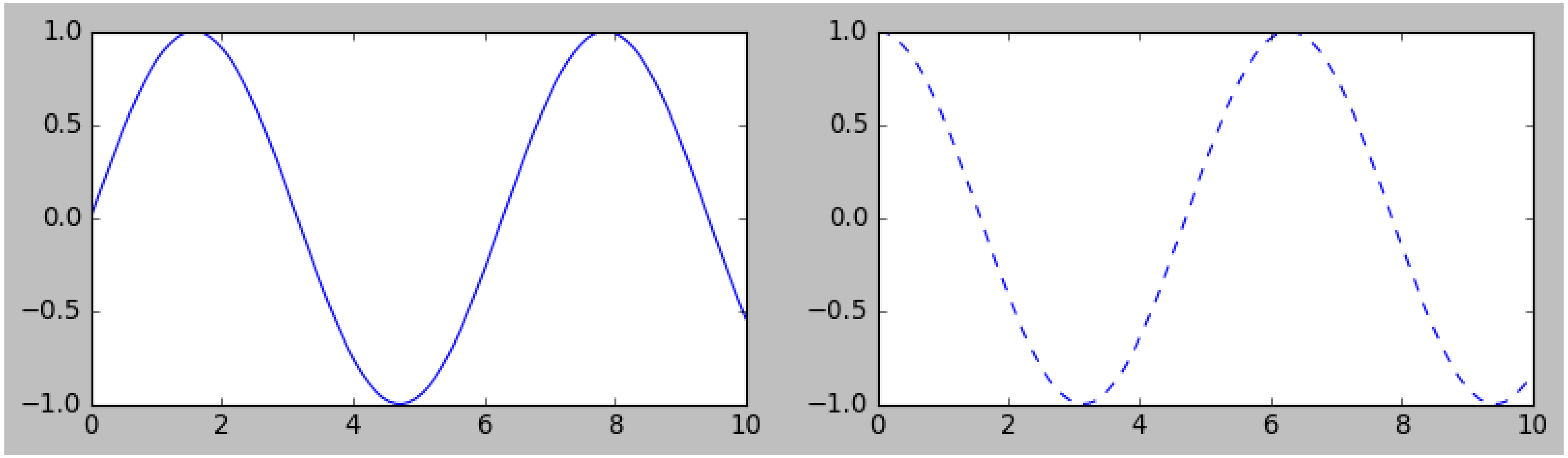
```
plt.figure()

plt.subplot(2,1,1)
plt.plot(x, np.sin(x),'-')

plt.subplot(2,1,2)
plt.plot(x, np.cos(x),'--');
```



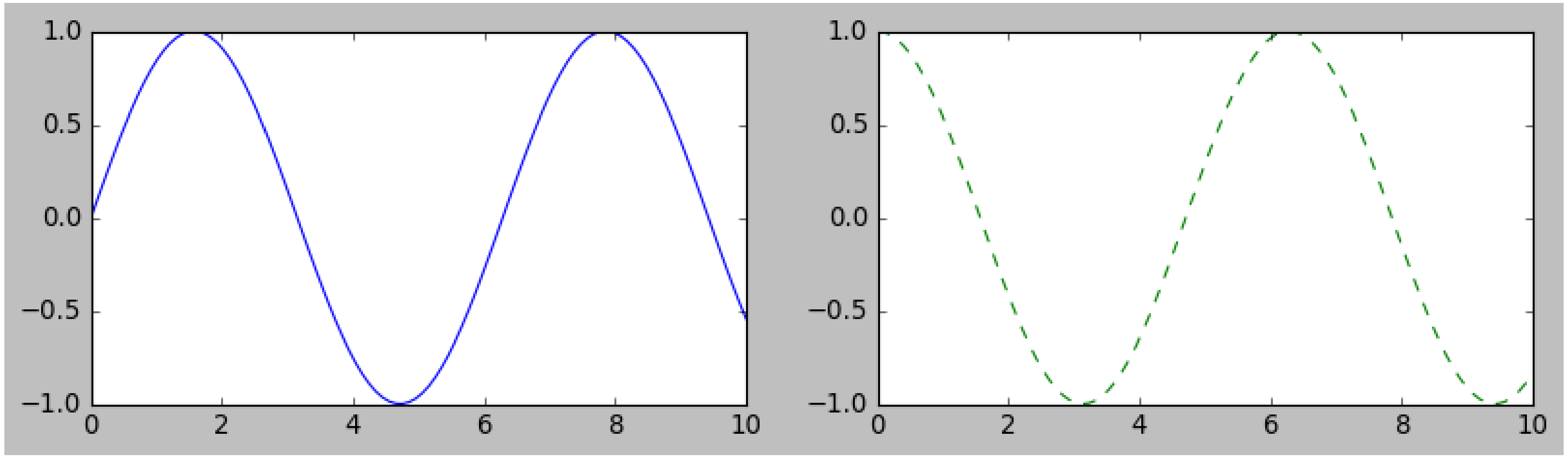
In [28]:
```
plt.figure()

plt.subplot(1,2,1)
plt.plot(x, np.sin(x),'-')

plt.subplot(1,2,2)
plt.plot(x, np.cos(x),'--');
```



In [31]:
```
plt.figure(figsize=(12,3))

plt.subplot(1,2,1)
plt.plot(x, np.sin(x),'-')

plt.subplot(1,2,2)
plt.plot(x, np.cos(x),'--');
```

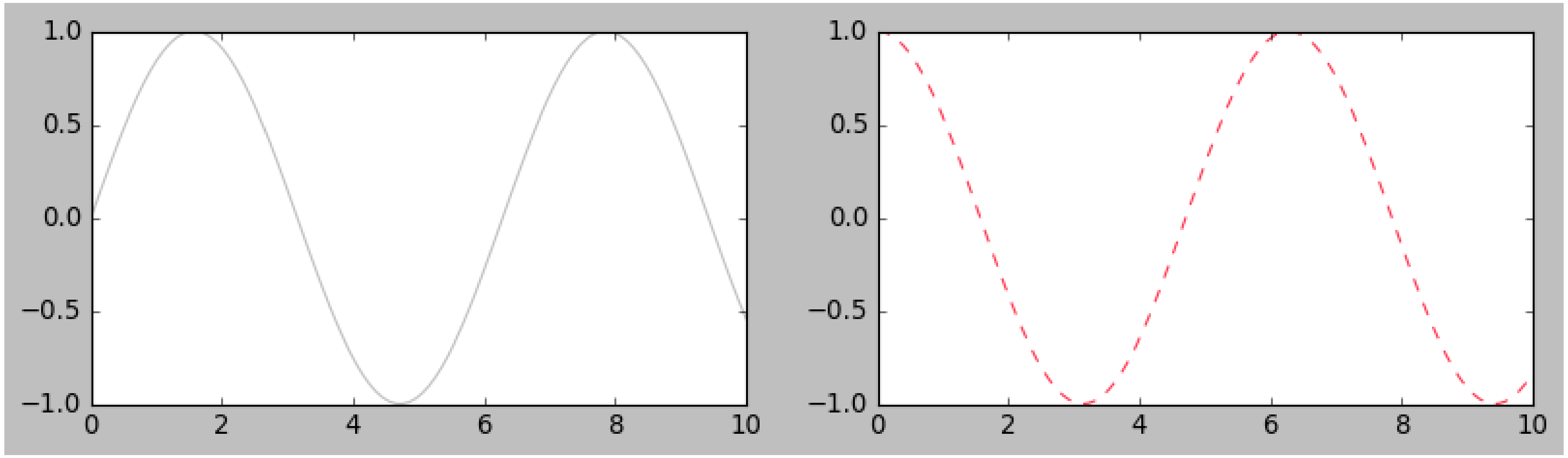


In [34]:
```
plt.figure(figsize=(12,3))

plt.subplot(1,2,1)
plt.plot(x, np.sin(x),'-', color='blue')

plt.subplot(1,2,2)
plt.plot(x, np.cos(x),'--',color='g');
```

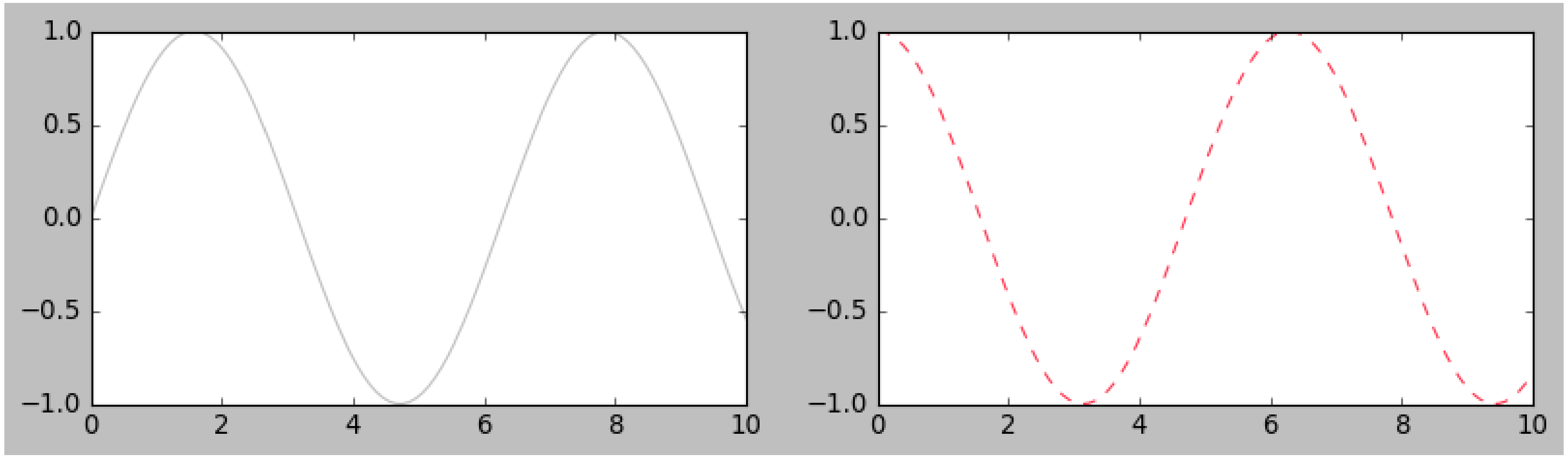


In [35]:

```
plt.figure(figsize=(12,3))

plt.subplot(1,2,1)
plt.plot(x, np.sin(x),'-', color='0.75') #0=dark

plt.subplot(1,2,2)
plt.plot(x, np.cos(x),'--',color=(1.0,0.2,0.3)); #RGB
```



In [36]:
```
plt.figure(figsize=(12,3))

plt.subplot(1,2,1)
plt.plot(x, np.sin(x),linestyle='-', color='0.75') #0=dark

plt.subplot(1,2,2)
plt.plot(x, np.cos(x),'--',color=(1.0,0.2,0.3)); #RGB
```

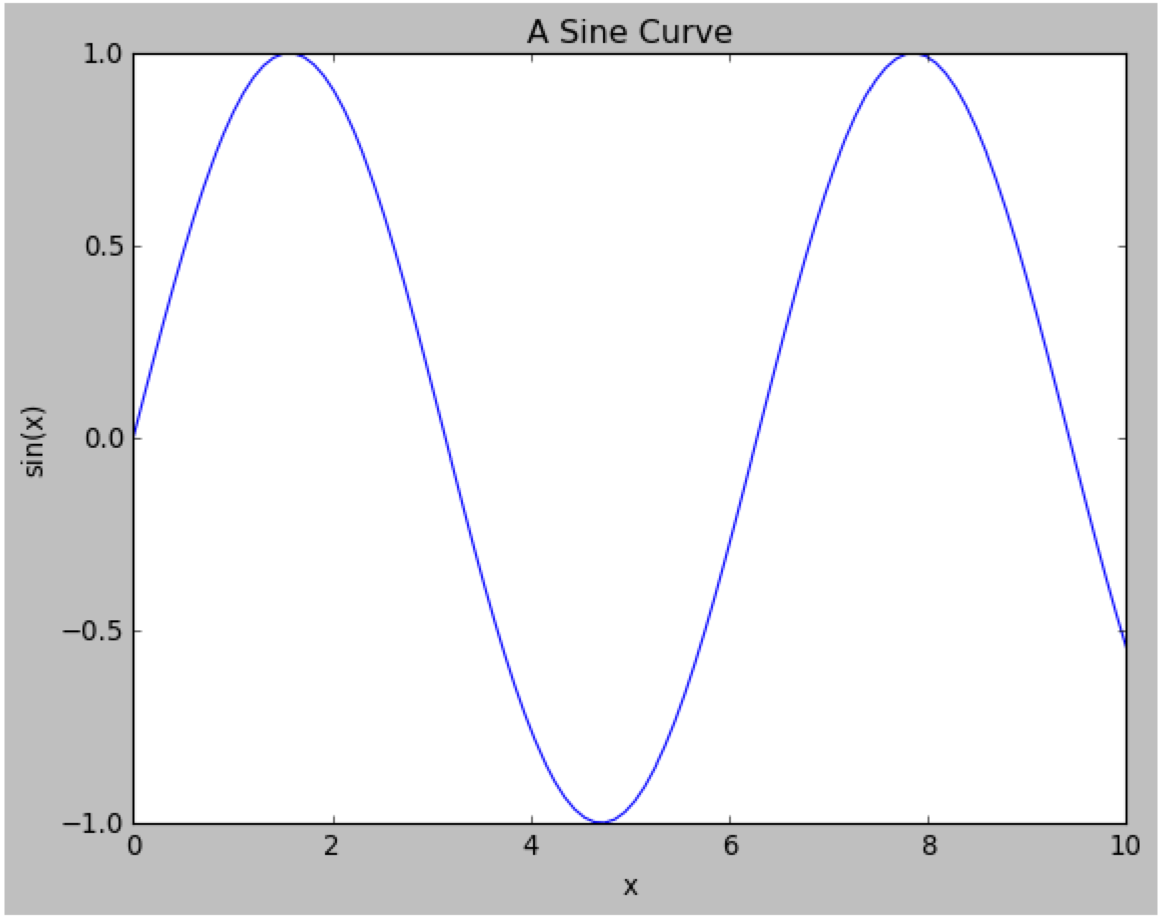


In [37]:
```
x
```

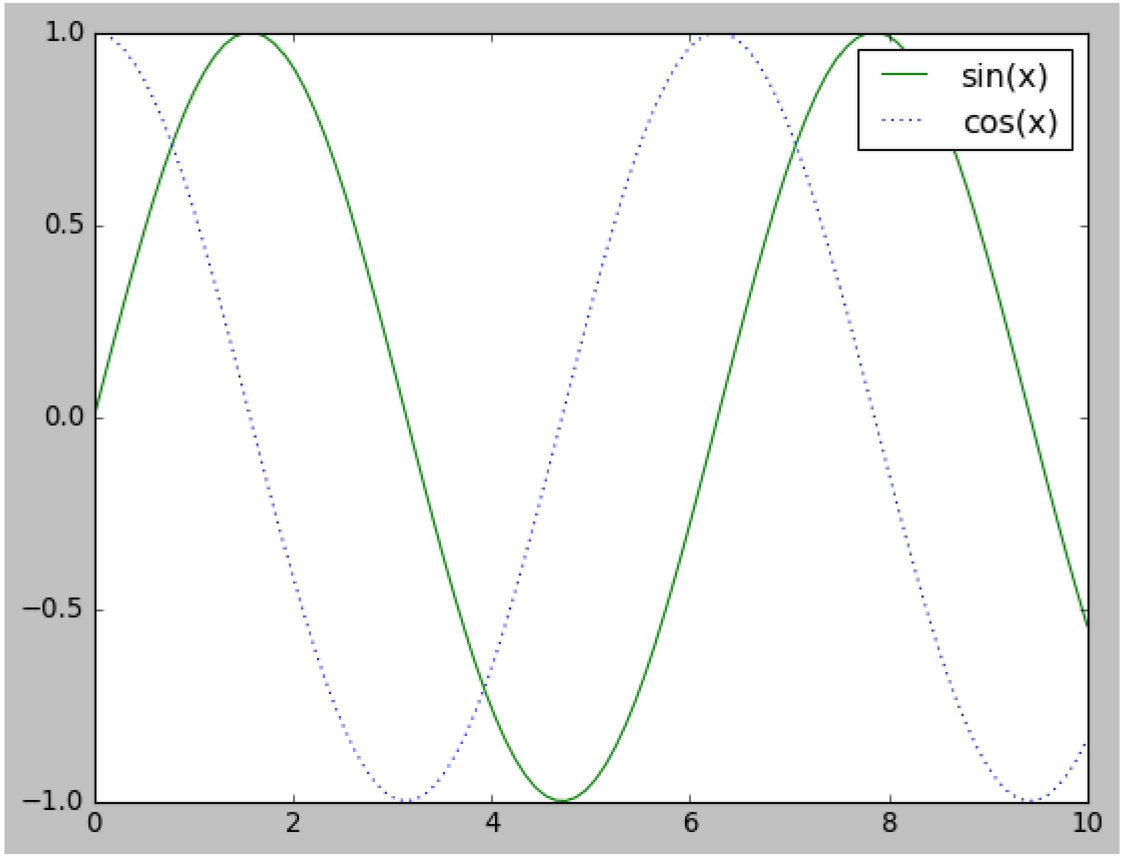
Out[37]:
```
array([ 0.        ,  0.1010101 ,  0.2020202 ,  0.3030303 ,  0.4040404 ,
        0.50505051,  0.60606061,  0.70707071,  0.80808081,  0.90909091,
        1.01010101,  1.11111111,  1.21212121,  1.31313131,  1.41414141,
        1.51515152,  1.61616162,  1.71717172,  1.81818182,  1.91919192,
        2.02020202,  2.12121212,  2.22222222,  2.32323232,  2.42424242,
        2.52525253,  2.62626263,  2.72727273,  2.82828283,  2.92929293,
        3.03030303,  3.13131313,  3.23232323,  3.33333333,  3.43434343,
        3.53535354,  3.63636364,  3.73737374,  3.83838384,  3.93939394,
        4.04040404,  4.14141414,  4.24242424,  4.34343434,  4.44444444,
        4.54545455,  4.64646465,  4.74747475,  4.84848485,  4.94949495,
        5.05050505,  5.15151515,  5.25252525,  5.35353535,  5.45454545,
        5.55555556,  5.65656566,  5.75757576,  5.85858586,  5.95959596,
        6.06060606,  6.16161616,  6.26262626,  6.36363636,  6.46464646,
        6.56565657,  6.66666667,  6.76767677,  6.86868687,  6.96969697,
        7.07070707,  7.17171717,  7.27272727,  7.37373737,  7.47474747,
        7.57575758,  7.67676768,  7.77777778,  7.87878788,  7.97979798,
        8.08080808,  8.18181818,  8.28282828,  8.38383838,  8.48484848,
        8.58585859,  8.68686869,  8.78787879,  8.88888889,  8.98989899,
        9.09090909,  9.19191919,  9.29292929,  9.39393939,  9.49494949,
        9.5959596 ,  9.6969697 ,  9.7979798 ,  9.8989899 , 10.        ])
```

In [41]:
```
# Label Plot

plt.plot(x, np.sin(x))
plt.title("A Sine Curve")
plt.xlabel("x")
plt.ylabel("sin(x)")
```

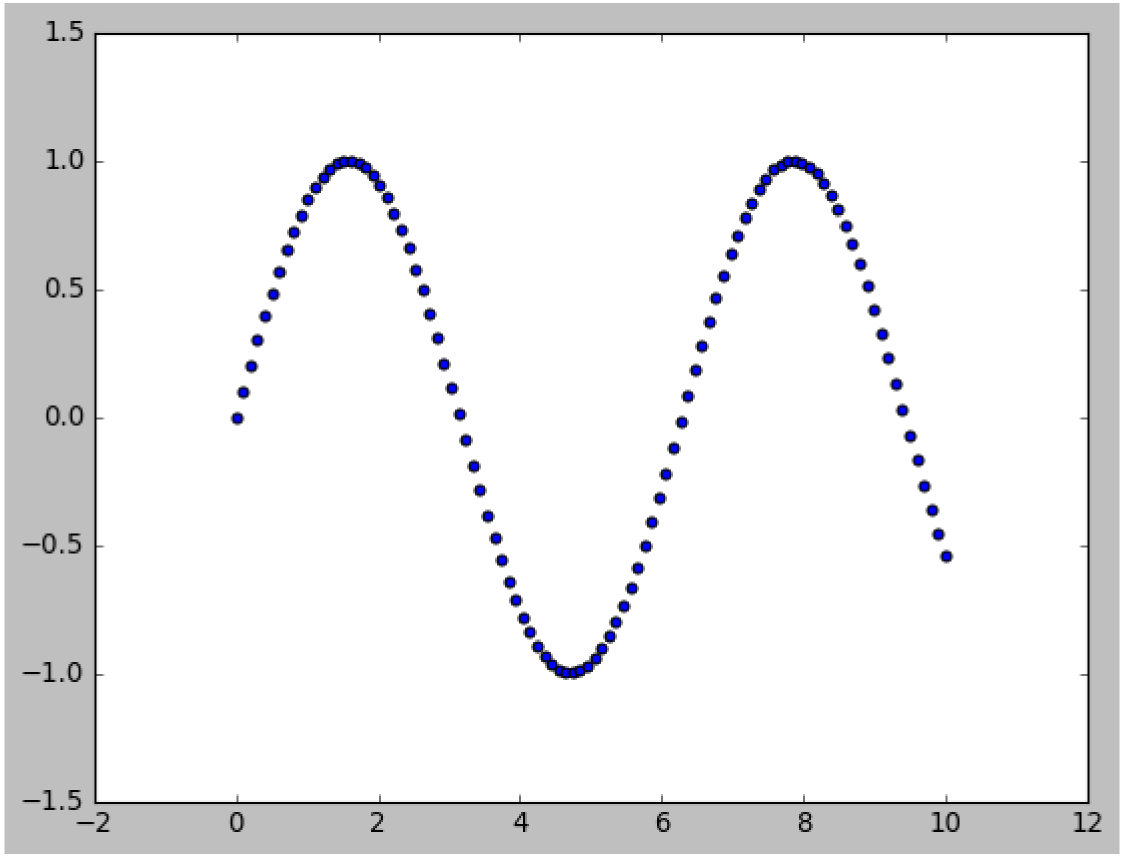
Out[41]:
```
Text(0, 0.5, 'sin(x)')
```



In [47]:
```
# Legend

plt.plot(x, np.sin(x), '-g',label='sin(x)')
plt.plot(x, np.cos(x), ':b', label='cos(x)')
plt.legend();
```

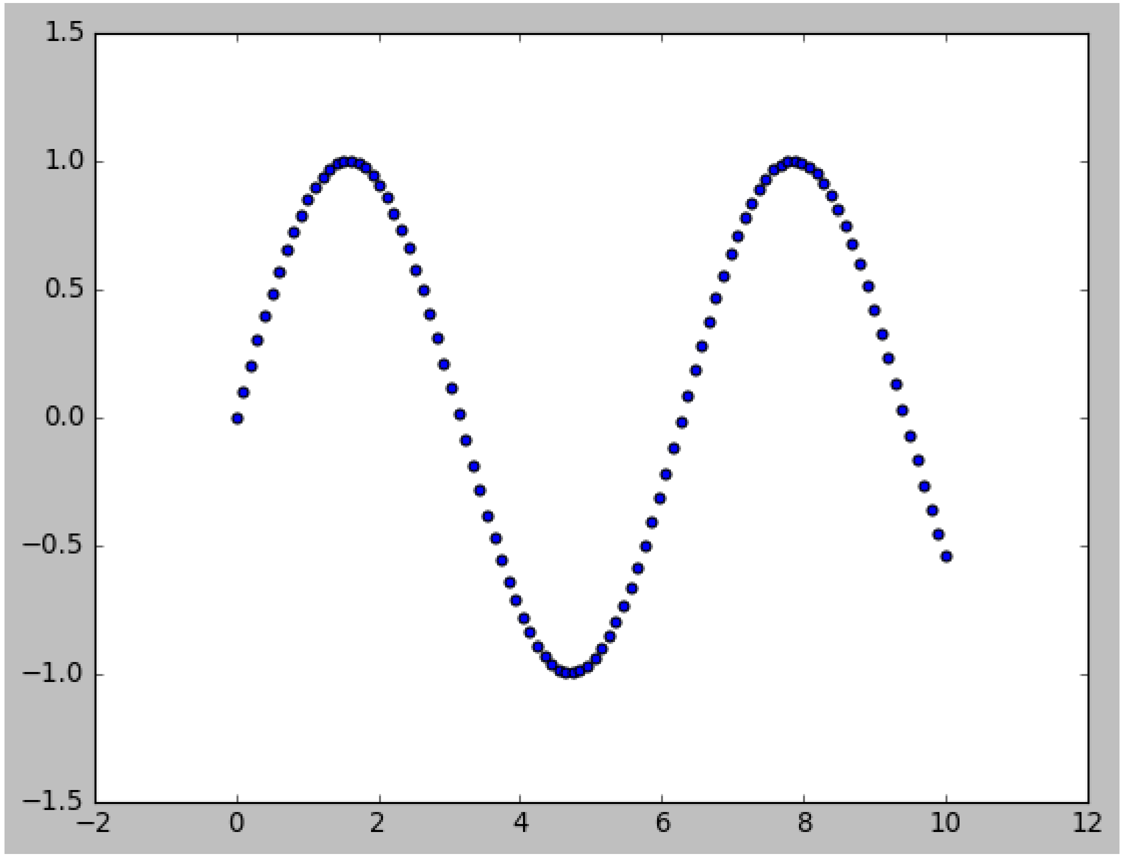


```
In [48]: # Scatter Plot

         plt.scatter(x, np.sin(x));
```

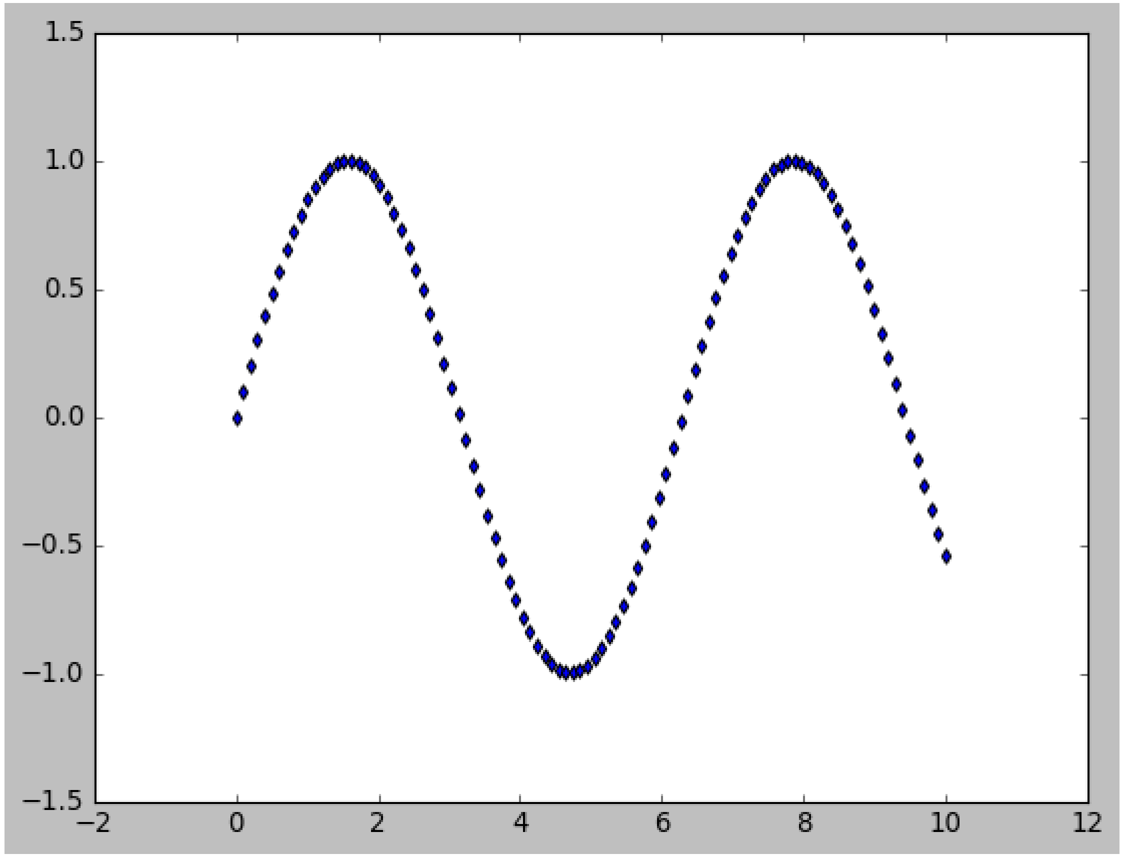


In [50]:

```
# Scatter Plot

plt.scatter(x, np.sin(x), marker='o');
```



In [56]:
```
# Scatter Plot

plt.scatter(x, np.sin(x), marker='d');
```



Seaborn

In [57]:
```
tips = sns.load_dataset('tips')
```

In [58]:
```
tips.head()
```

Out[58]:

	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 [59]:
```
tips.sex.value_counts()
```

Out[59]:
```
Male      157
Female     87
Name: sex, dtype: int64
```

In [60]:
```
tips.smoker.value_counts()
```

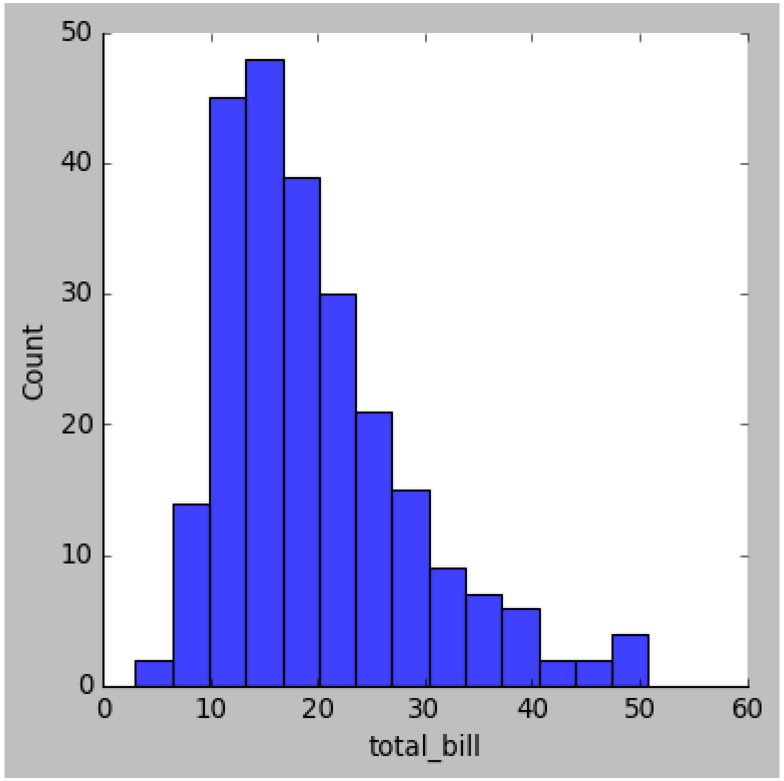
Out[60]:
```
No     151
Yes     93
Name: smoker, dtype: int64
```

In [61]:
```
tips.day.value_counts()
```

Out[61]:
```
Sat     87
Sun     76
Thur    62
Fri     19
Name: day, dtype: int64
```

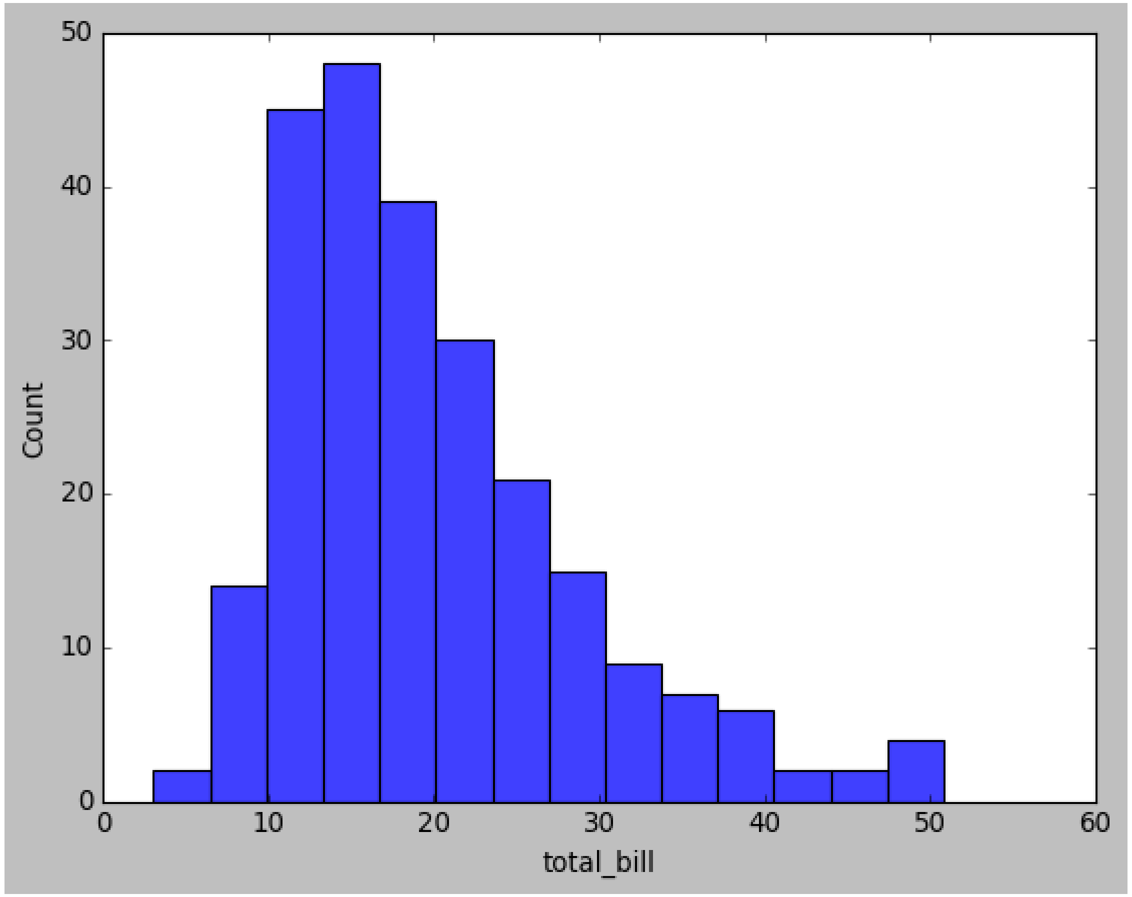
In [65]:

```
# distplot()

sns.displot(tips['total_bill']);
```



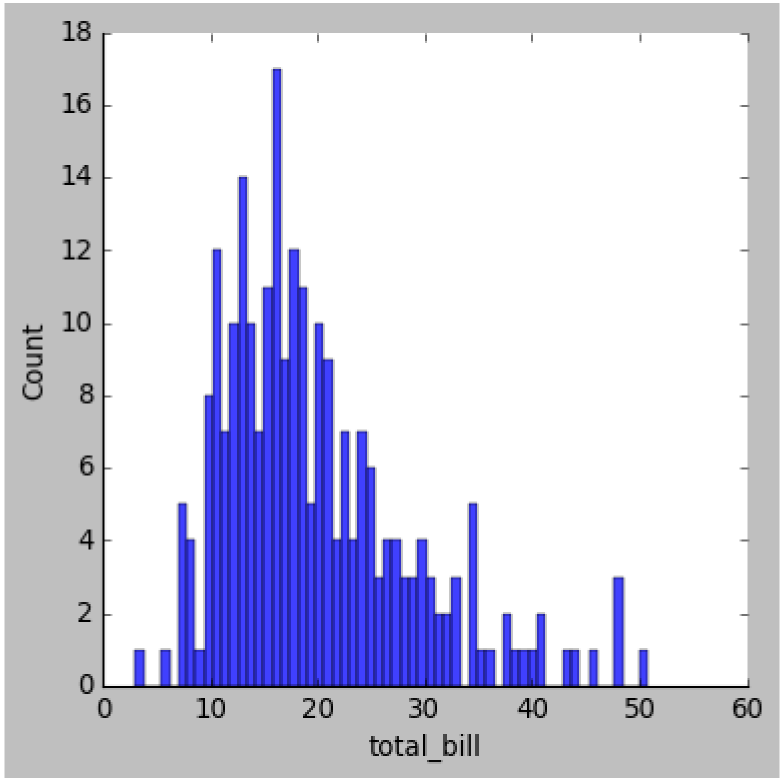
In [66]:

```
# distplot()

sns.histplot(tips['total_bill']);
```

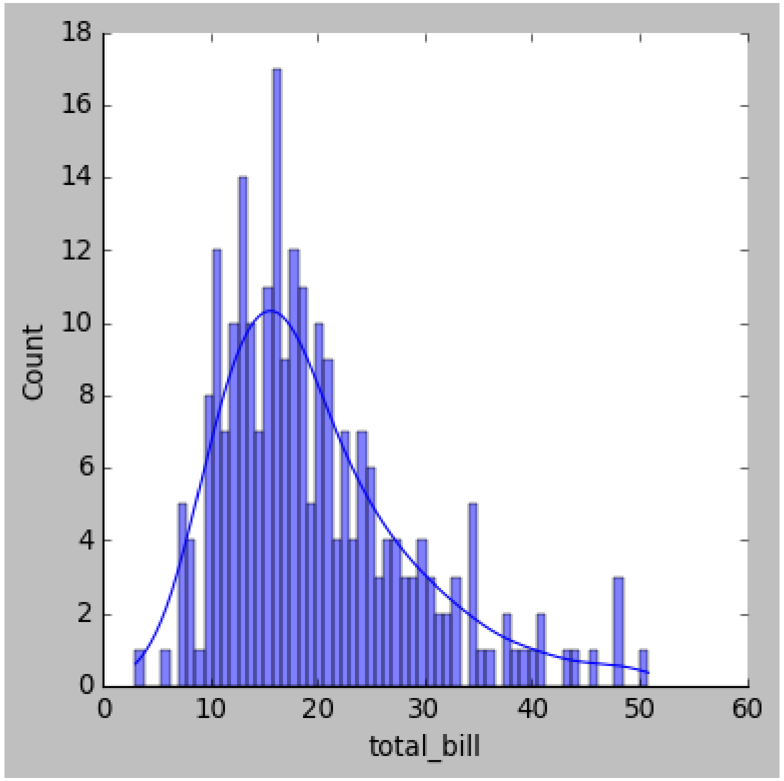


In [67]:

```
# distplot()

sns.displot(tips['total_bill'], bins=60);
```



```
In [68]: # distplot()

         sns.displot(tips['total_bill'], bins=60, kde=True);
```

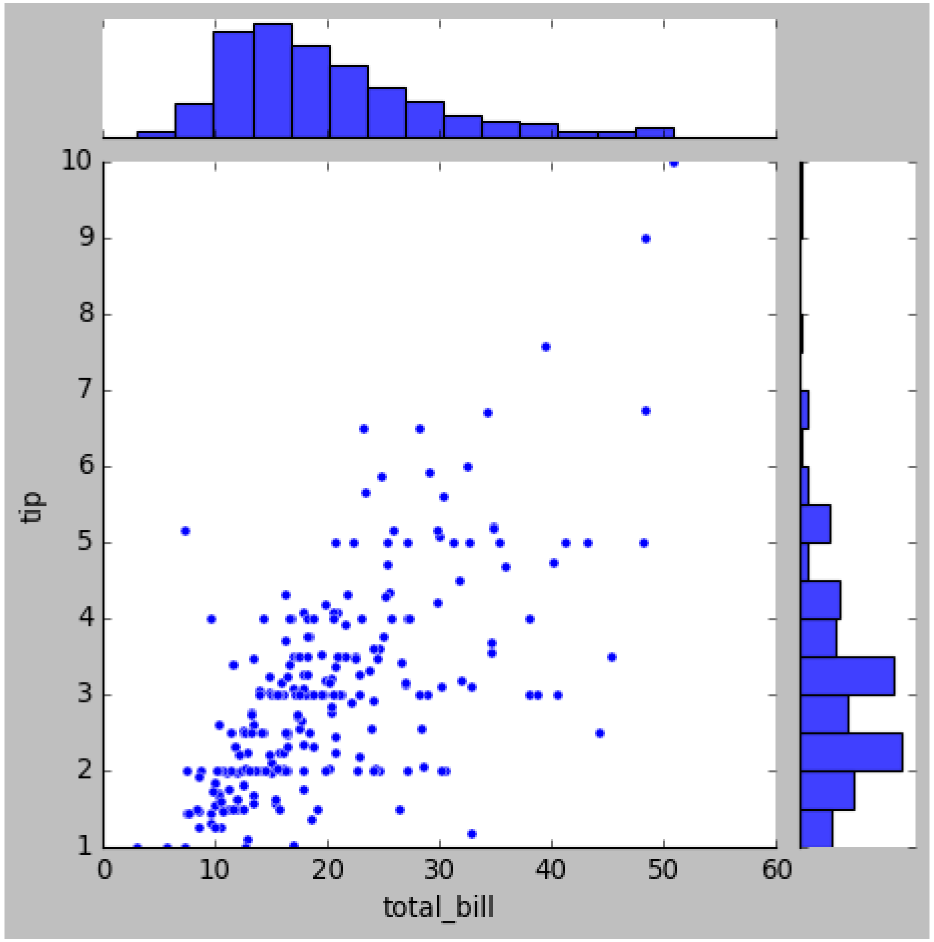


In [71]:
```
tips.head()
```

Out[71]:

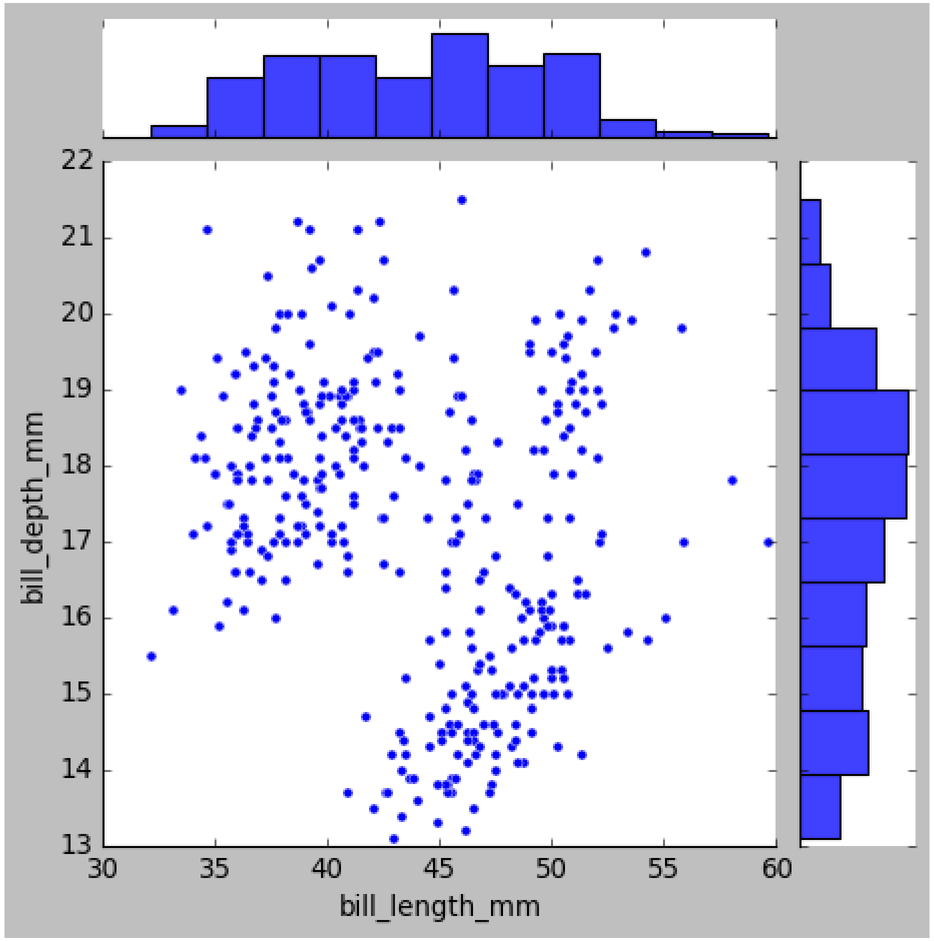
	total_bill	tip	sex	smoker	day	time	size
0	16.99	1.01	Female	No	Sun	Dinner	2
1	10.34	1.66	Male	No	Sun	Dinner	3
2	21.01	3.50	Male	No	Sun	Dinner	3
3	23.68	3.31	Male	No	Sun	Dinner	2
4	24.59	3.61	Female	No	Sun	Dinner	4

In [78]:
```
sns.jointplot(x='total_bill', y='tip', data=tips, kind='scatter');
```

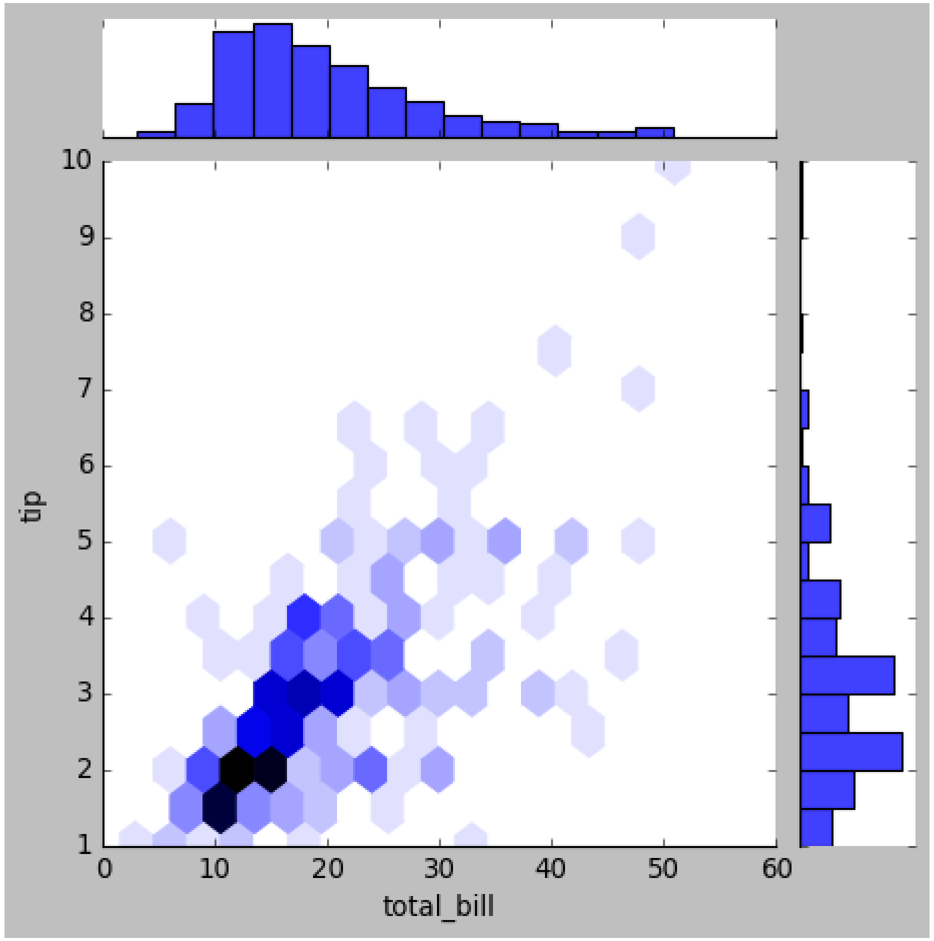


```
In [76]: penguins = sns.load_dataset("penguins")
         sns.jointplot(data=penguins, x="bill_length_mm", y="bill_depth_mm")
```

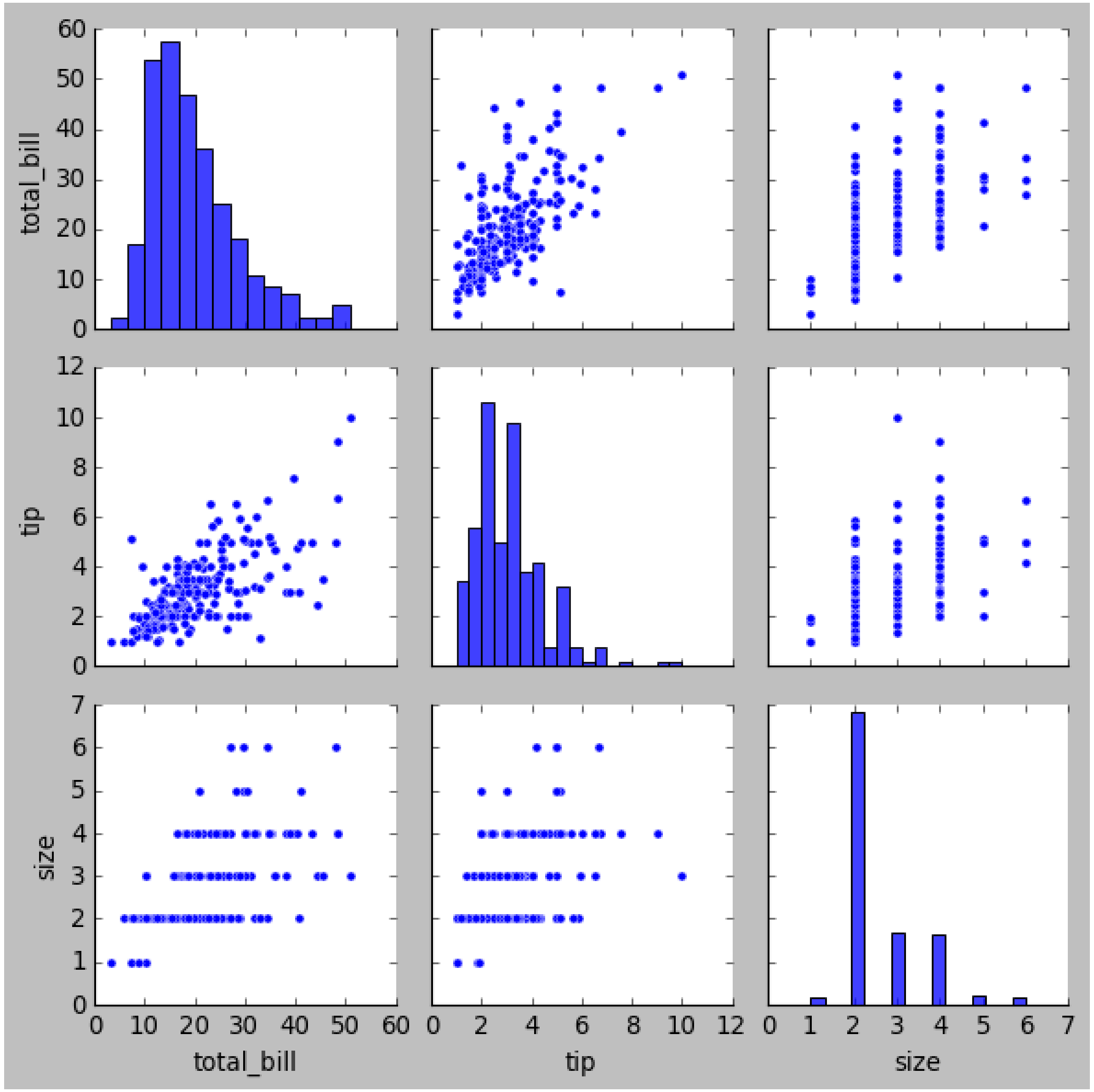
```
Out[76]: <seaborn.axisgrid.JointGrid at 0x20586fe5550>
```



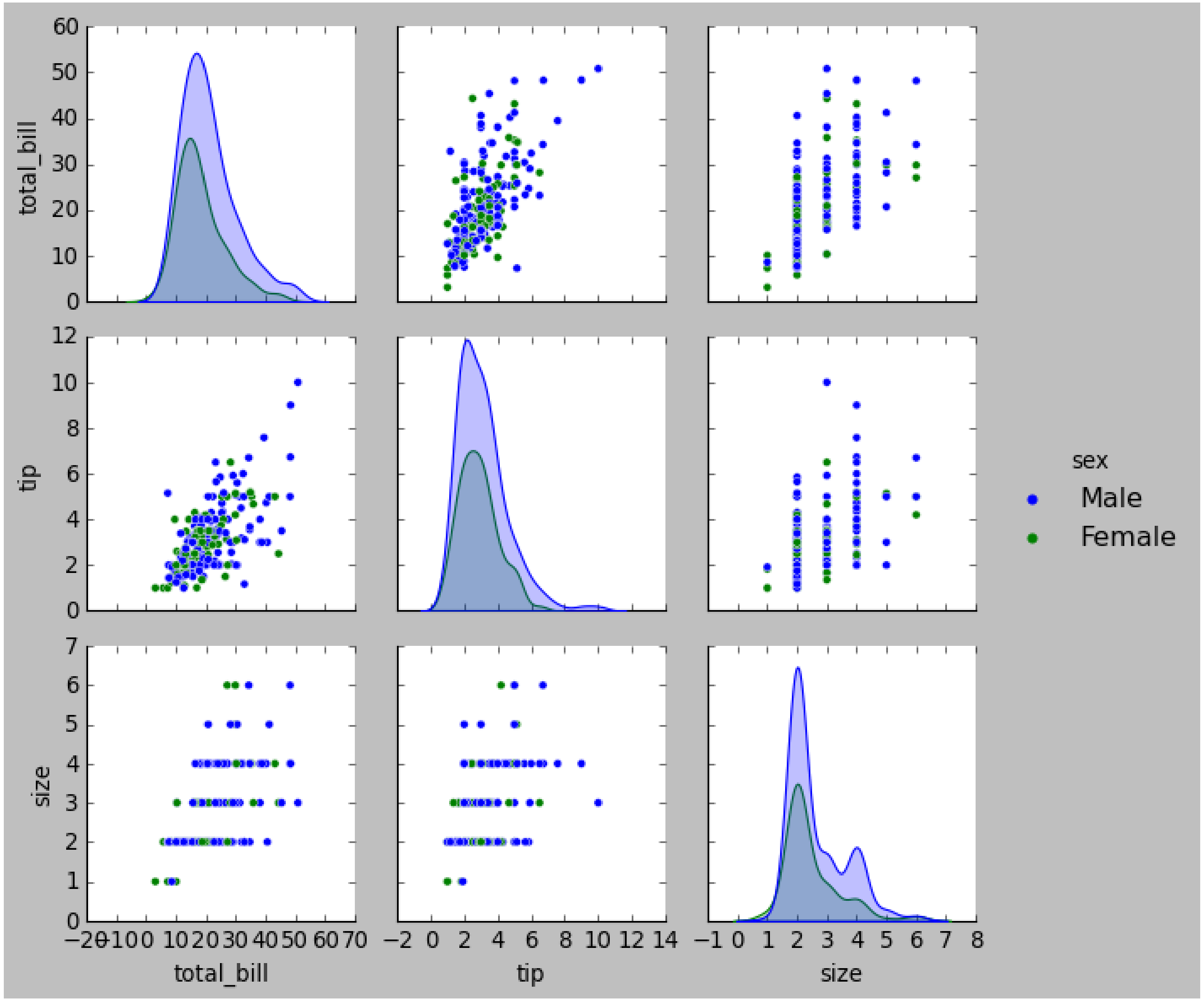
```
In [79]: sns.jointplot(x='total_bill', y='tip', data=tips, kind='hex');
```



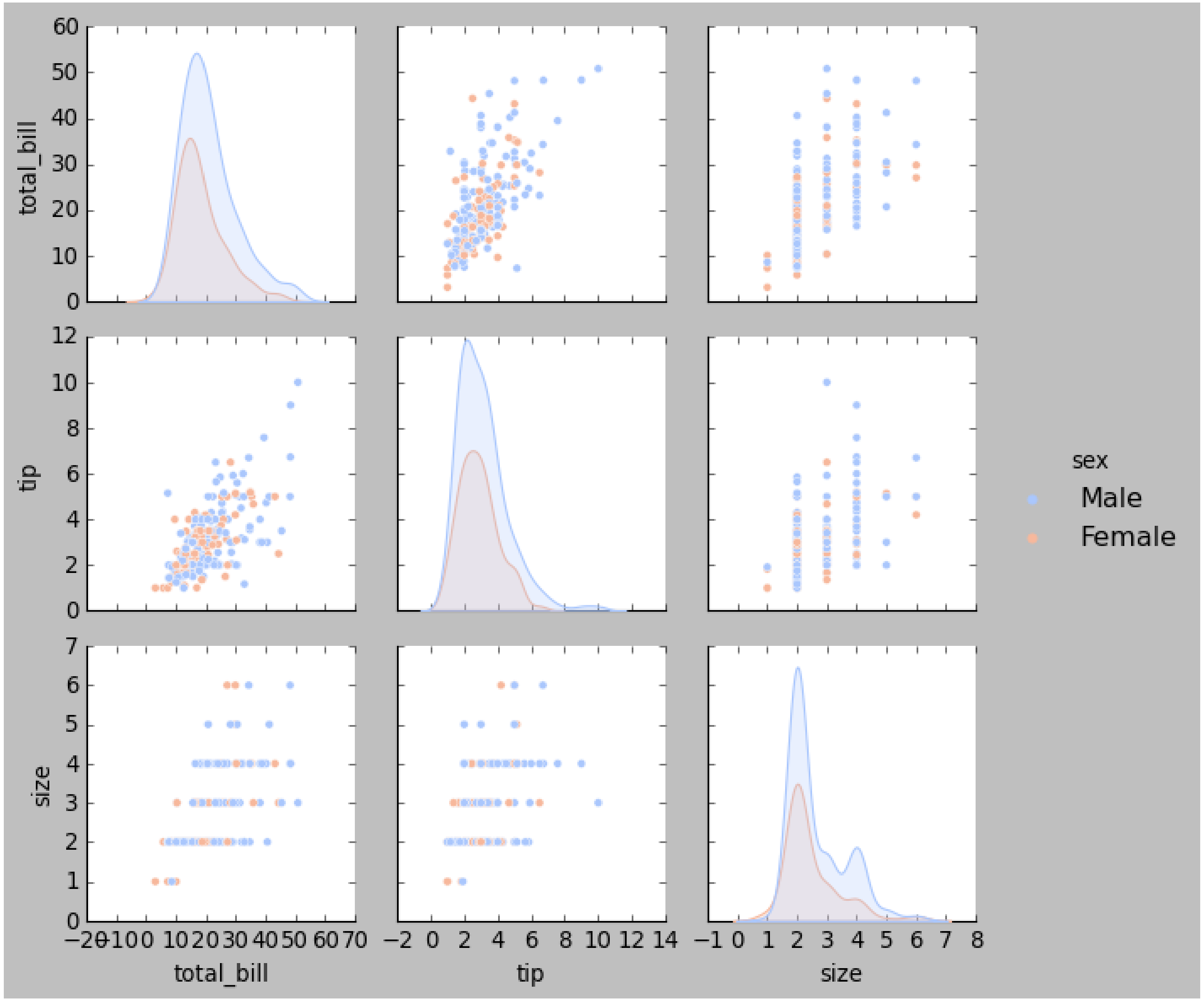
In [80]:
```
sns.pairplot(tips);sns.pairplot(tips);
```



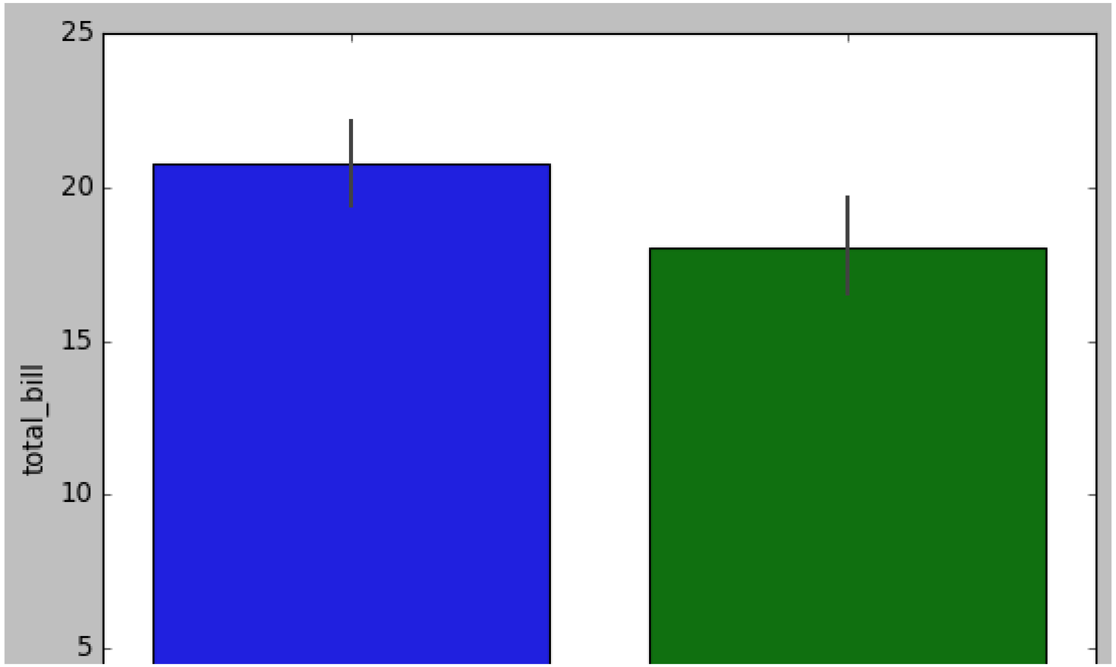
In [81]:
```
sns.pairplot(tips, hue='sex');
```



```
In [82]: sns.pairplot(tips, hue='sex', palette='coolwarm');
```

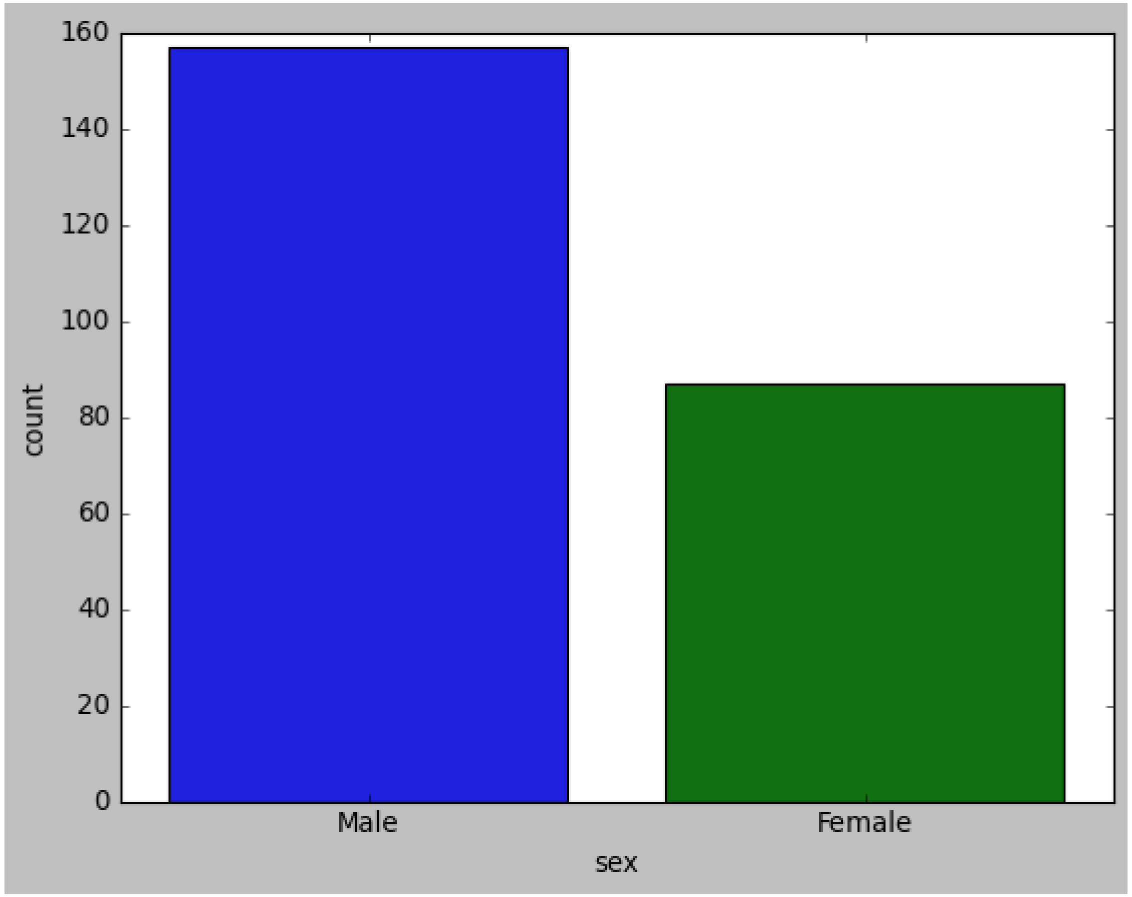


In [83]:
```
sns.barplot(x='sex', y='total_bill', data=tips);
```

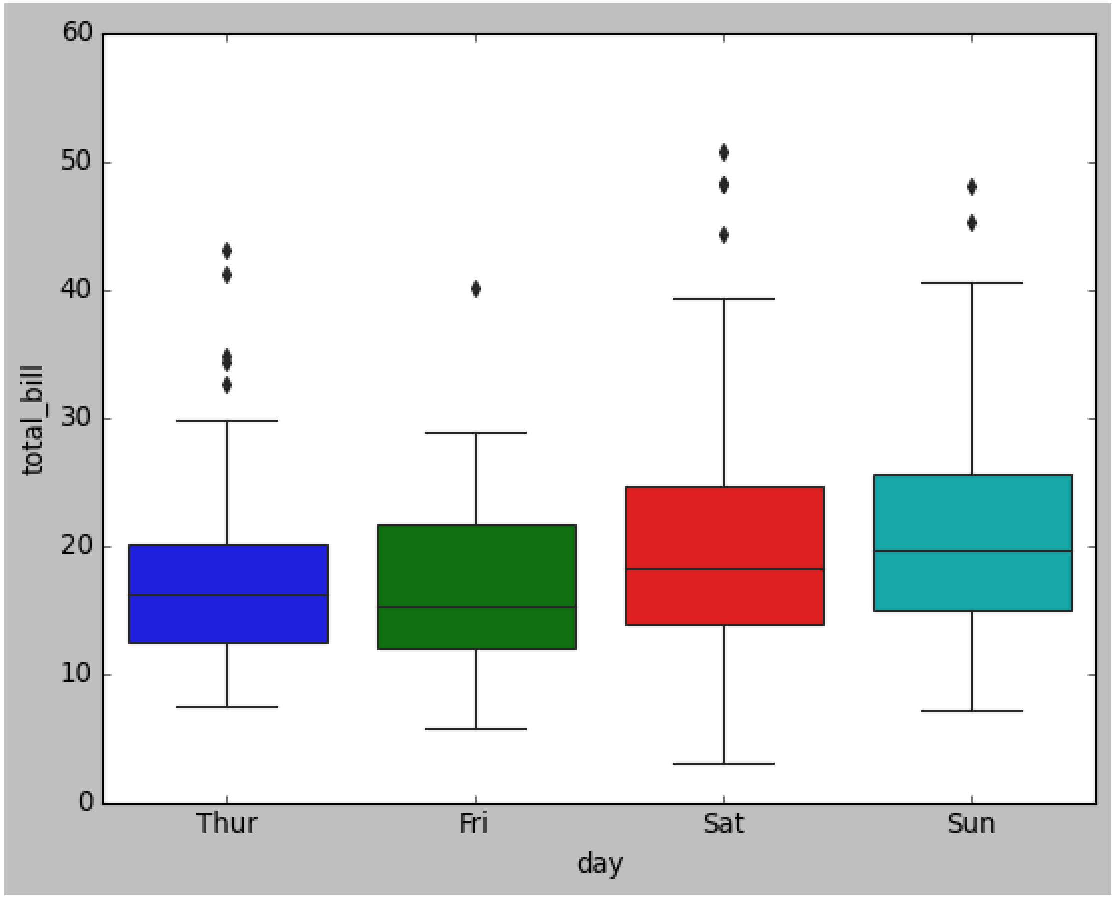


In [84]:
```
sns.countplot(x='sex', data=tips)
```

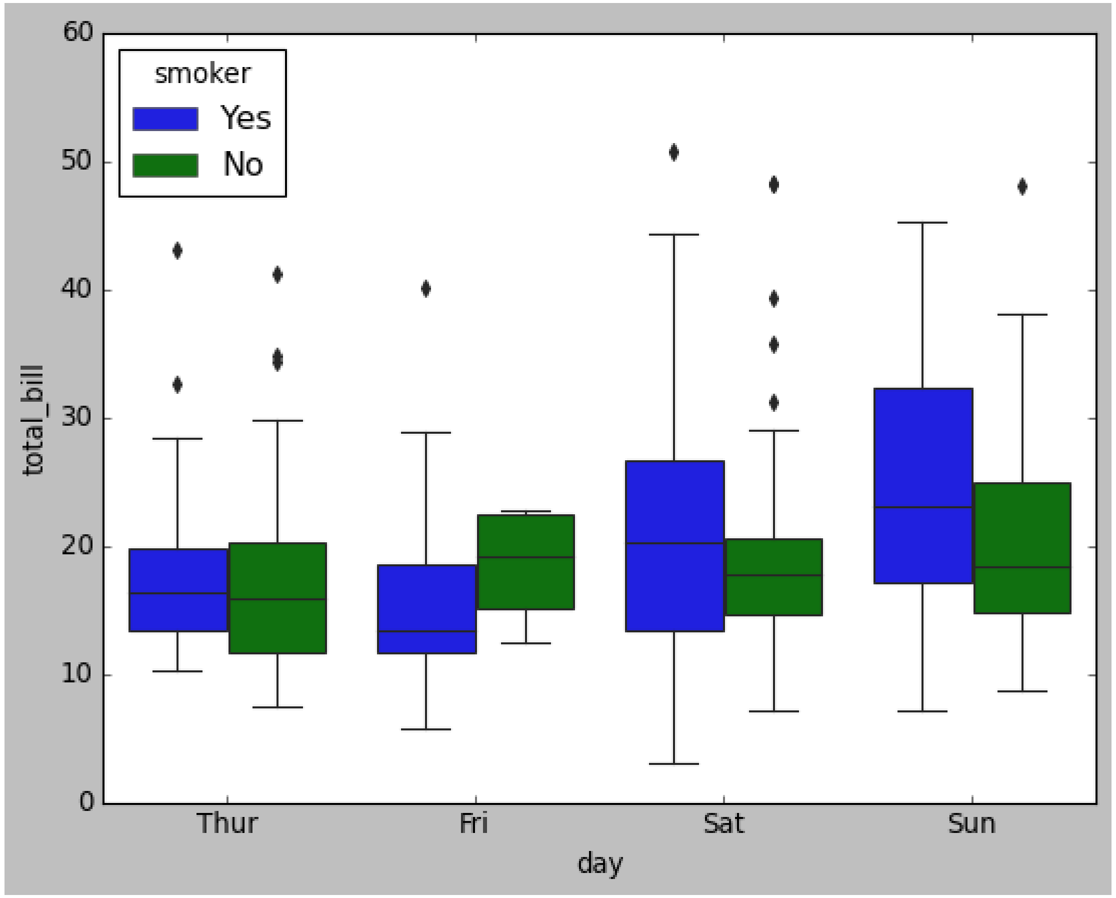
Out[84]:
```
<AxesSubplot:xlabel='sex', ylabel='count'>
```



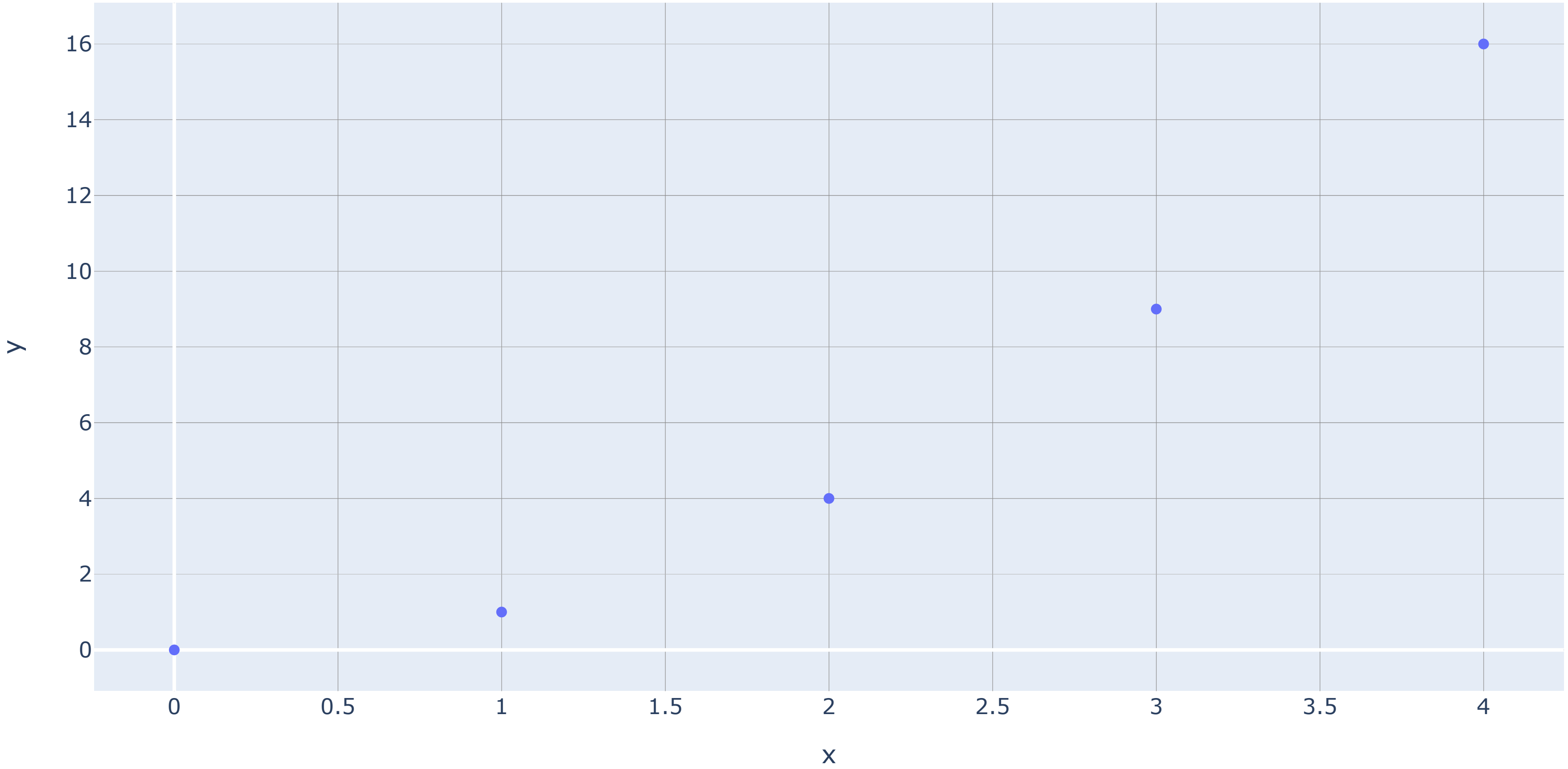
```
In [85]: sns.boxplot(x='day', y='total_bill', data=tips);
```



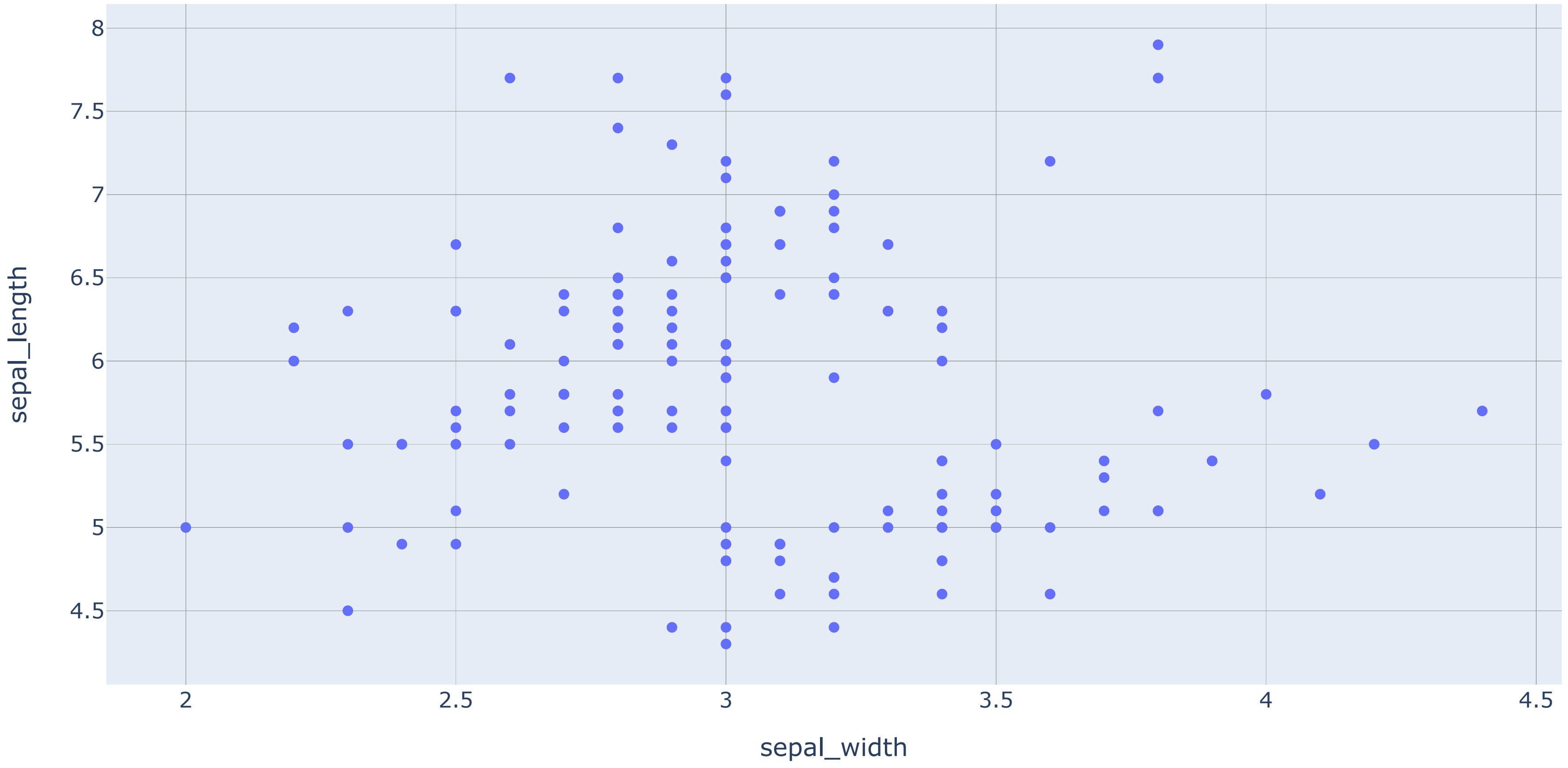
```
In [87]: sns.boxplot(x='day', y='total_bill', data=tips, hue='smoker');
```



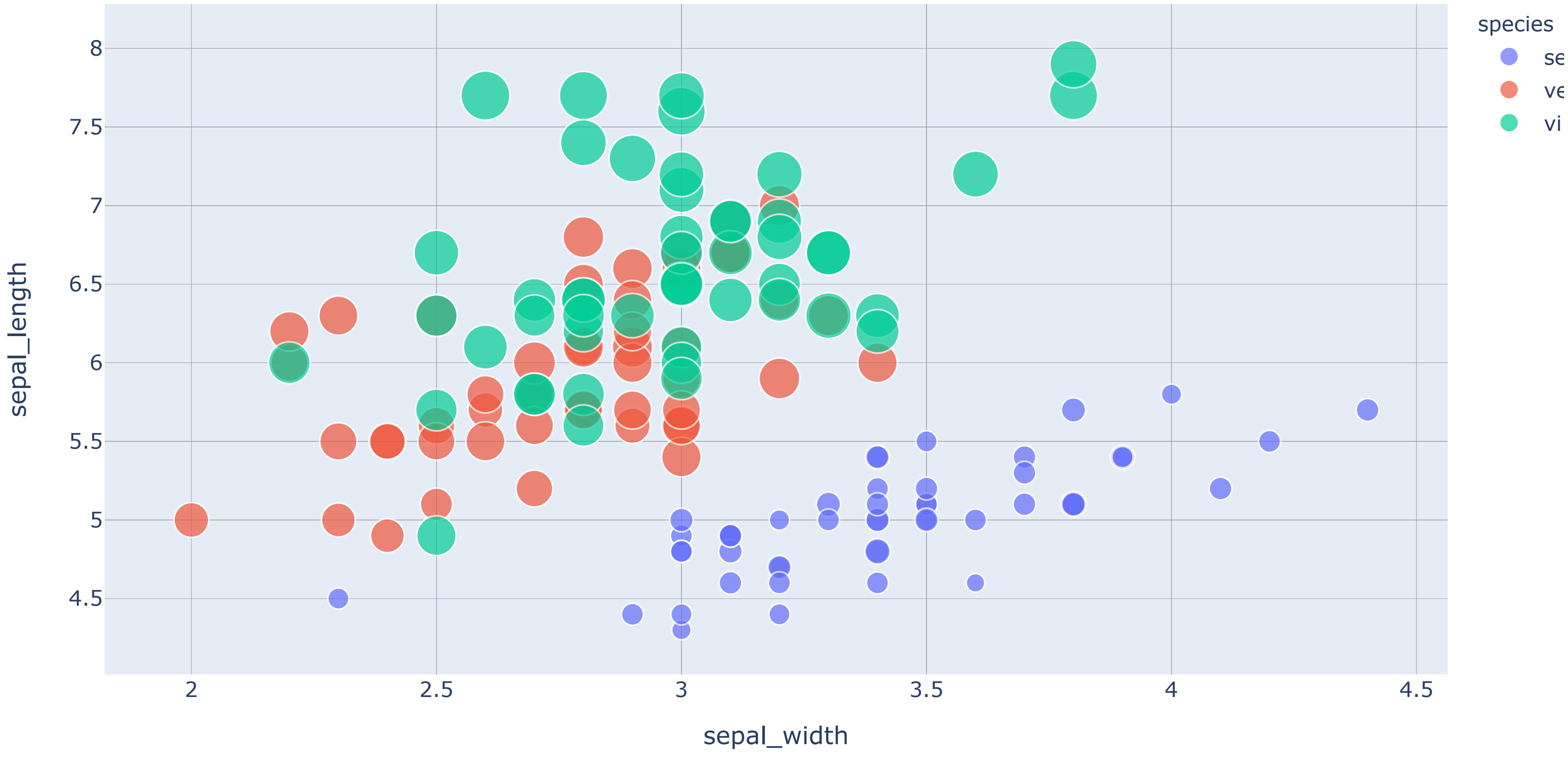
```
In [88]: # x and y given as array_like objects
         import plotly.express as px
         fig = px.scatter(x=[0, 1, 2, 3, 4], y=[0, 1, 4, 9, 16])
         fig.show()
```



```
In [89]: df = px.data.iris() # iris is a pandas DataFrame
         fig = px.scatter(df, x="sepal_width", y="sepal_length")
         fig.show()
```

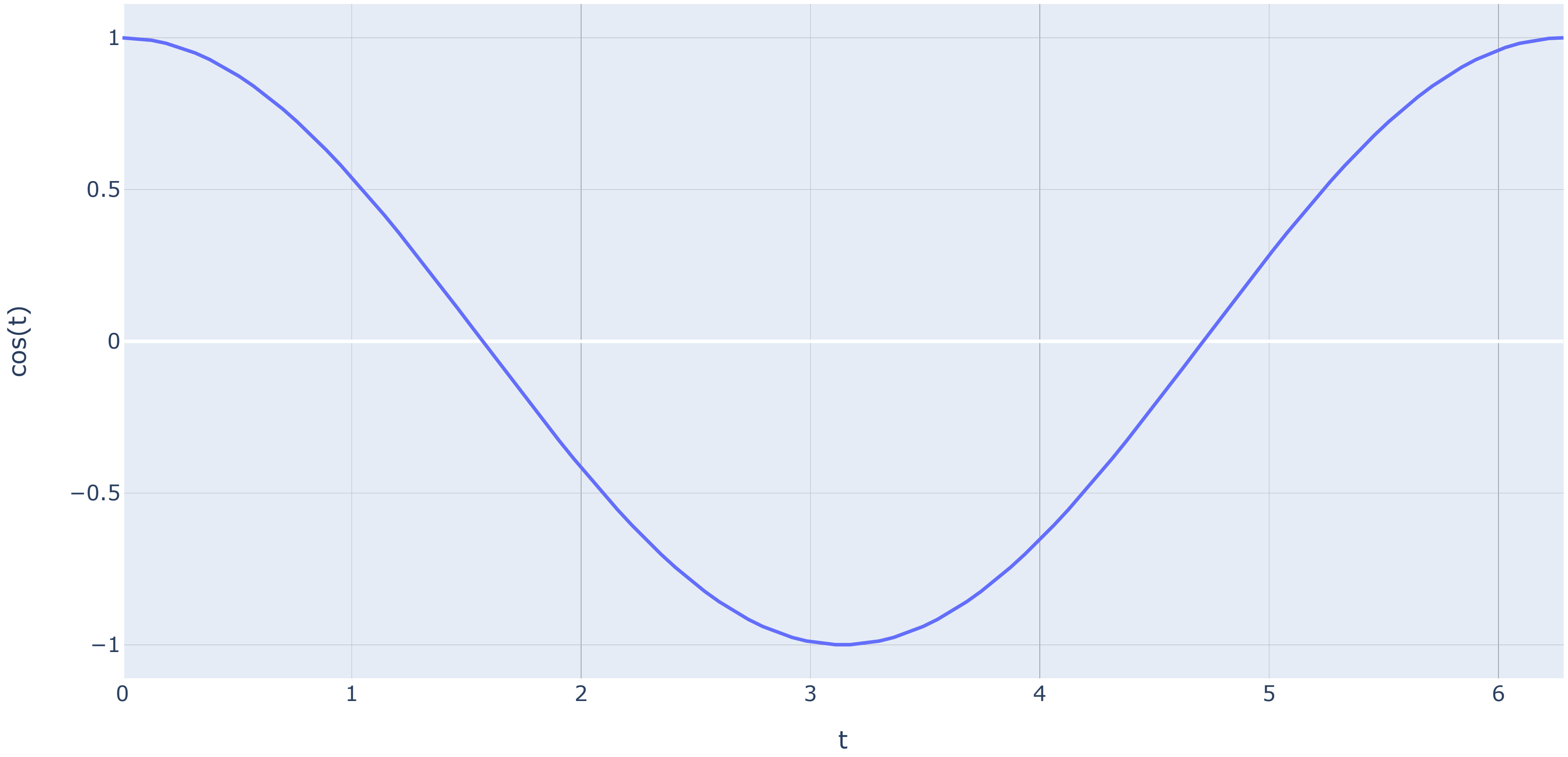


In [90]:
```
df = px.data.iris()
fig = px.scatter(df, x="sepal_width", y="sepal_length", color="species",
                 size='petal_length', hover_data=['petal_width'])
fig.show()
```



In [91]:

```
t = np.linspace(0, 2*np.pi, 100)

fig = px.line(x=t, y=np.cos(t), labels={'x':'t', 'y':'cos(t)'})
fig.show()
```



In [ ]:

```
```