

```
In [2]: import pandas as pd
import plotly as px
import numpy as np
import plotly.express as pxe
pd.options.plotting.backend = "plotly"

tips = pd.read_csv(r'C:\Users\Master\Desktop\Jupyter\tips.csv')
tips.describe()
```

Out[2]:

	total_bill	tip	size
count	244.000000	244.000000	244.000000
mean	19.785943	2.998279	2.569672
std	8.902412	1.383638	0.951100
min	3.070000	1.000000	1.000000
25%	13.347500	2.000000	2.000000
50%	17.795000	2.900000	2.000000
75%	24.127500	3.562500	3.000000
max	50.810000	10.000000	6.000000

```
In [3]: tips
```

Out[3]:

	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
...
239	29.03	5.92	Male	No	Sat	Dinner	3
240	27.18	2.00	Female	Yes	Sat	Dinner	2
241	22.67	2.00	Male	Yes	Sat	Dinner	2
242	17.82	1.75	Male	No	Sat	Dinner	2
243	18.78	3.00	Female	No	Thur	Dinner	2

```
In [4]: np.unique(tips['sex'],return_counts=True)
```

Out[4]: (array(['Female', 'Male'], dtype=object), array([87, 157], dtype=int64))

```
In [5]: tips.isnull().sum()
```

Out[5]:

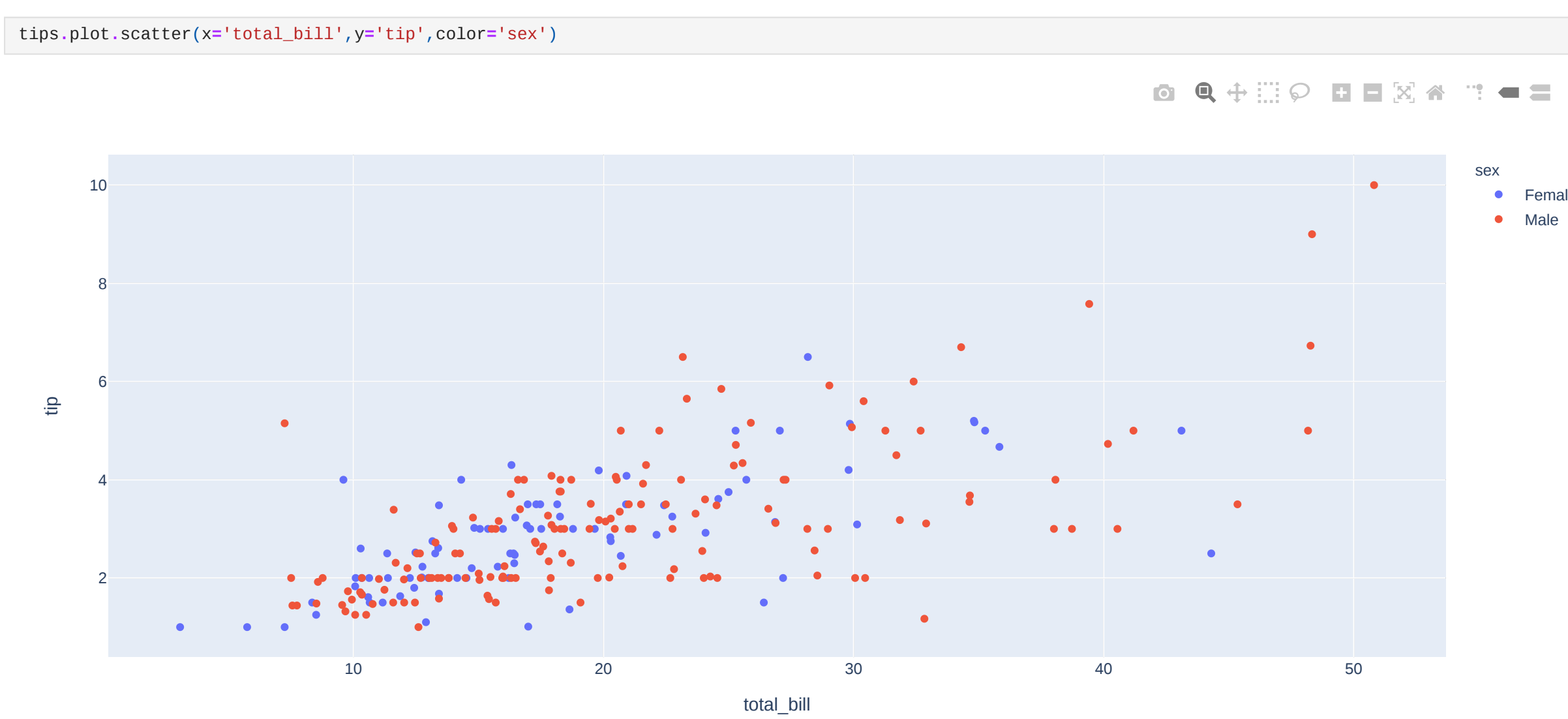
total_bill	0
tip	0
sex	0
smoker	0
day	0
time	0
size	0
dtype:	int64

```
In [6]: tips['%'] = (tips['tip'] / tips['total_bill']) * 100
```

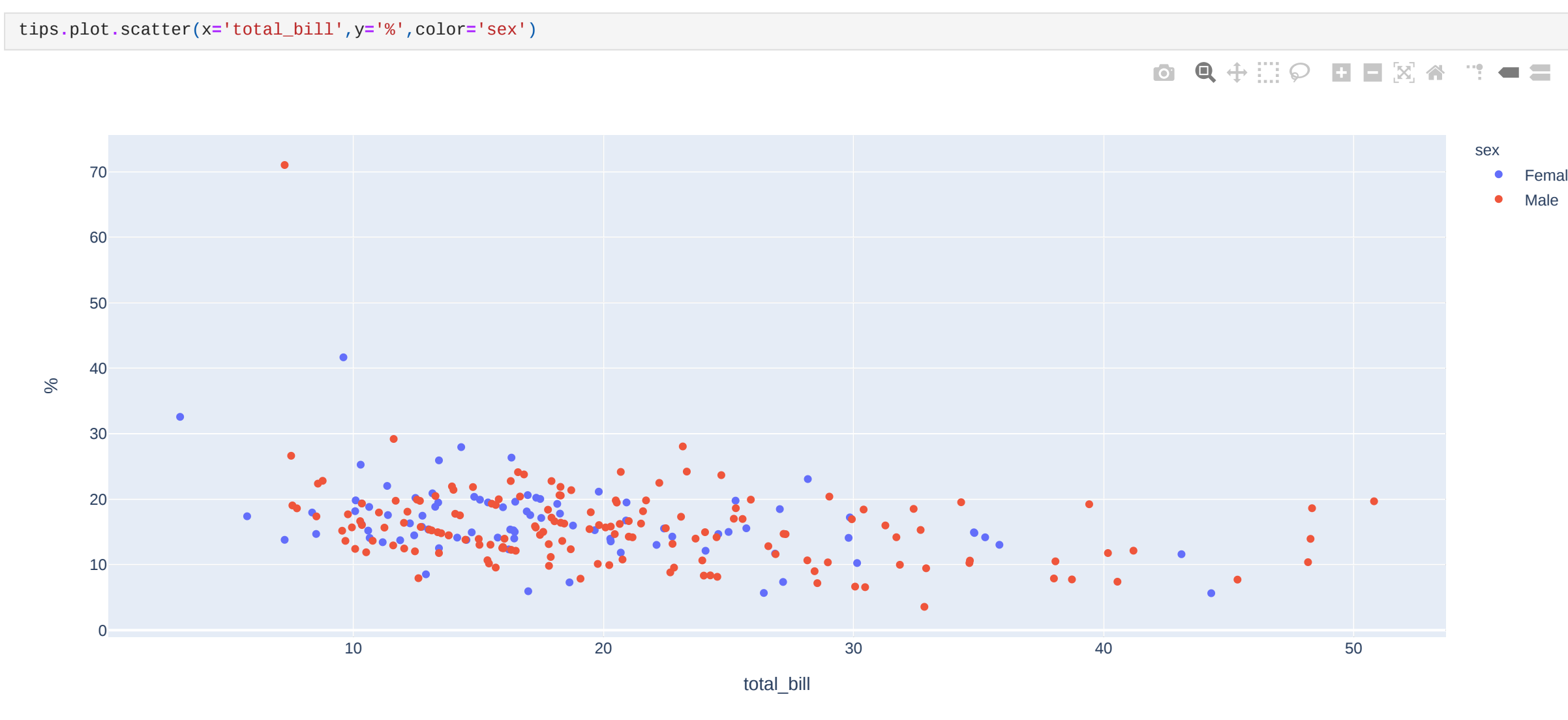
Out[6]:

	total_bill	tip	sex	smoker	day	time	size	%
0	16.99	1.01	Female	No	Sun	Dinner	2	5.944673
1	10.34	1.66	Male	No	Sun	Dinner	3	16.054159
2	21.01	3.50	Male	No	Sun	Dinner	3	16.658734
3	23.68	3.31	Male	No	Sun	Dinner	2	13.978041
4	24.59	3.61	Female	No	Sun	Dinner	4	14.680705
...
239	29.03	5.92	Male	No	Sat	Dinner	3	20.392697
240	27.18	2.00	Female	Yes	Sat	Dinner	2	7.358352
241	22.67	2.00	Male	Yes	Sat	Dinner	2	8.822232
242	17.82	1.75	Male	No	Sat	Dinner	2	9.820426
243	18.78	3.00	Female	No	Thur	Dinner	2	15.974441

```
In [7]: tips.plot.scatter(x='total_bill',y='tip',color='sex')
```



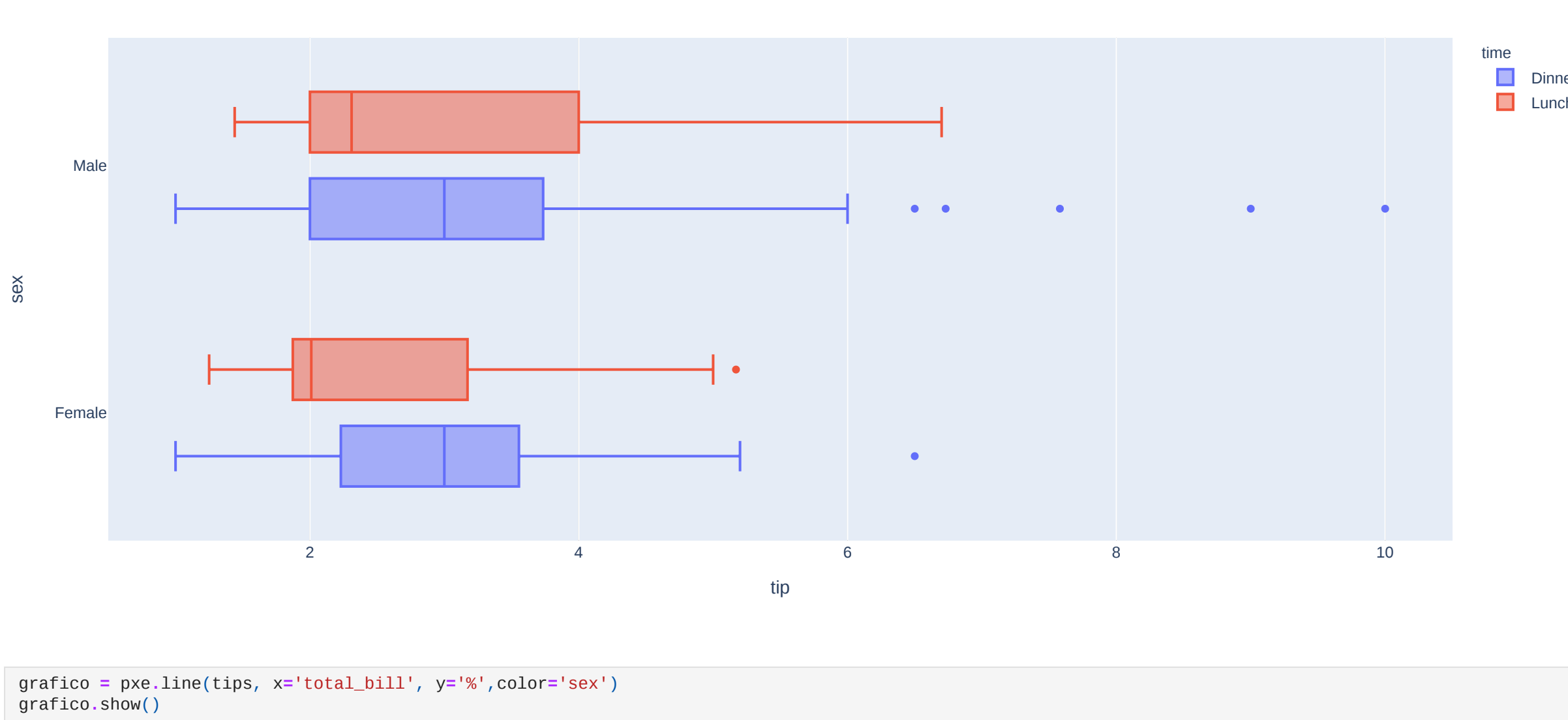
```
In [29]: tips.plot.scatter(x='total_bill',y='%',color='sex')
```



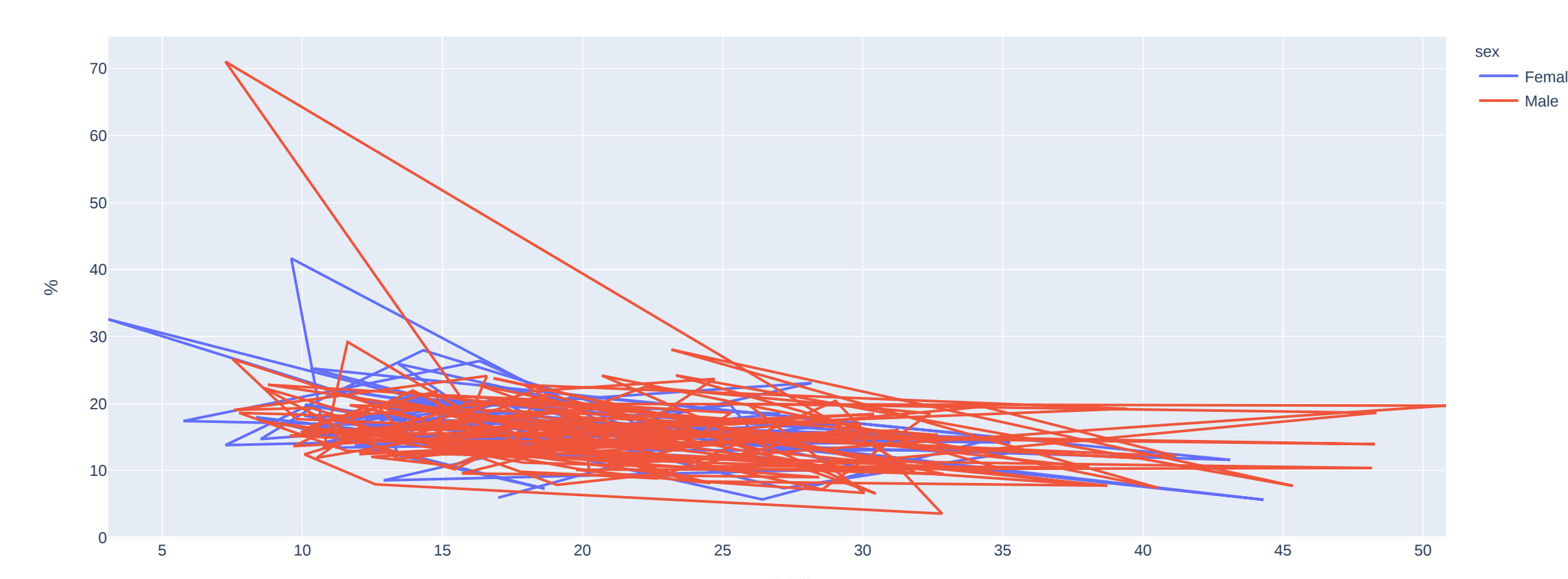
```
In [8]: grafico = pxe.scatter_matrix(tips, dimensions=['sex','%','total_bill','time','smoker'],color= 'sex')
grafico.show()
```



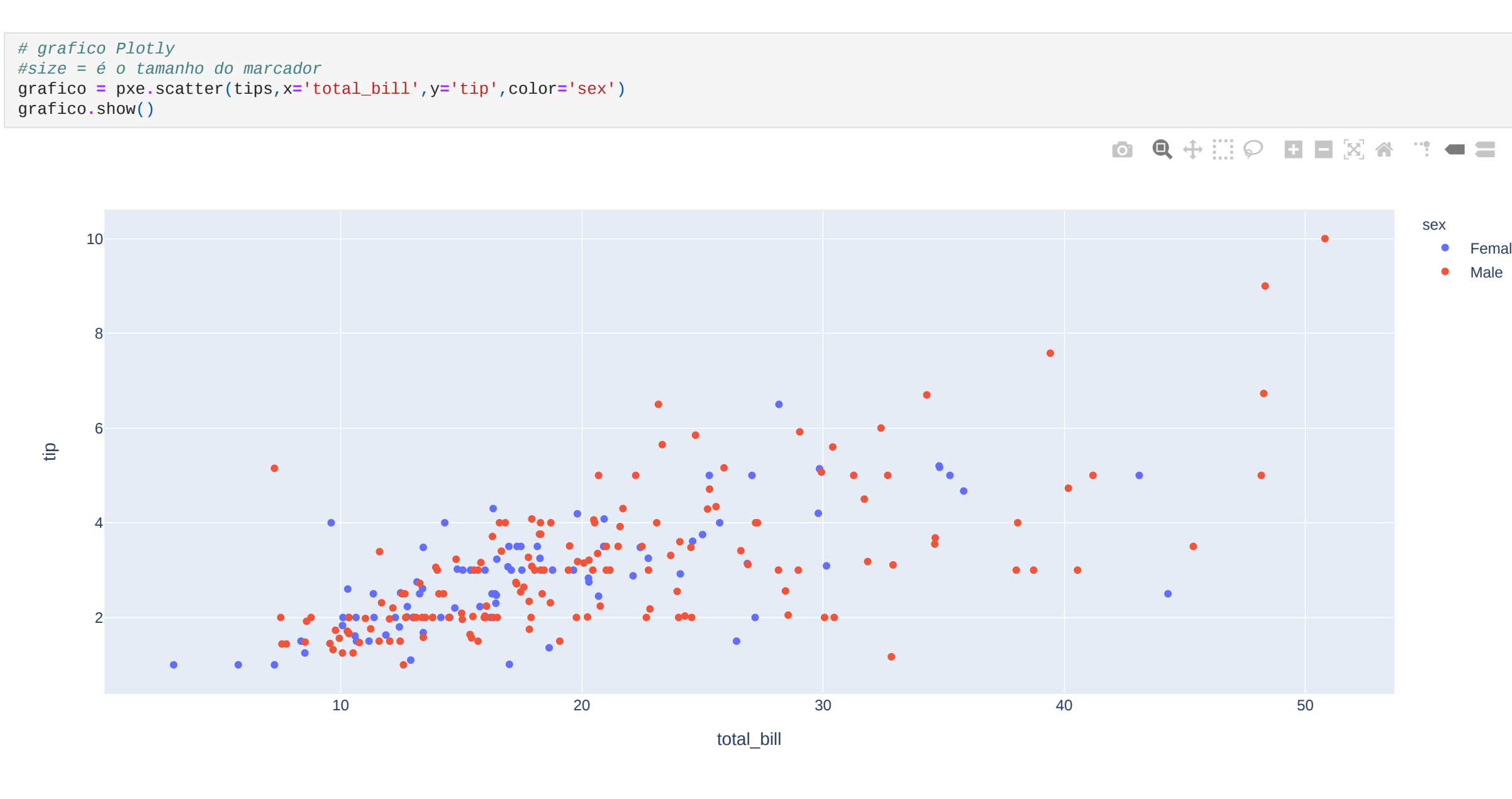
```
In [8]: tips.plot.box(x='tip', y ='sex',color='time')
```



```
In [12]: grafico = pxe.line(tips, x='total_bill', y='%',color='sex')
grafico.show()
```



```
In [31]: # grafico Plotly
#size = é o tamanho do marcador
#size = é o tamanho do marcador
grafico = pxe.scatter(tips,x='total_bill',y='tip',color='sex')
grafico.show()
```



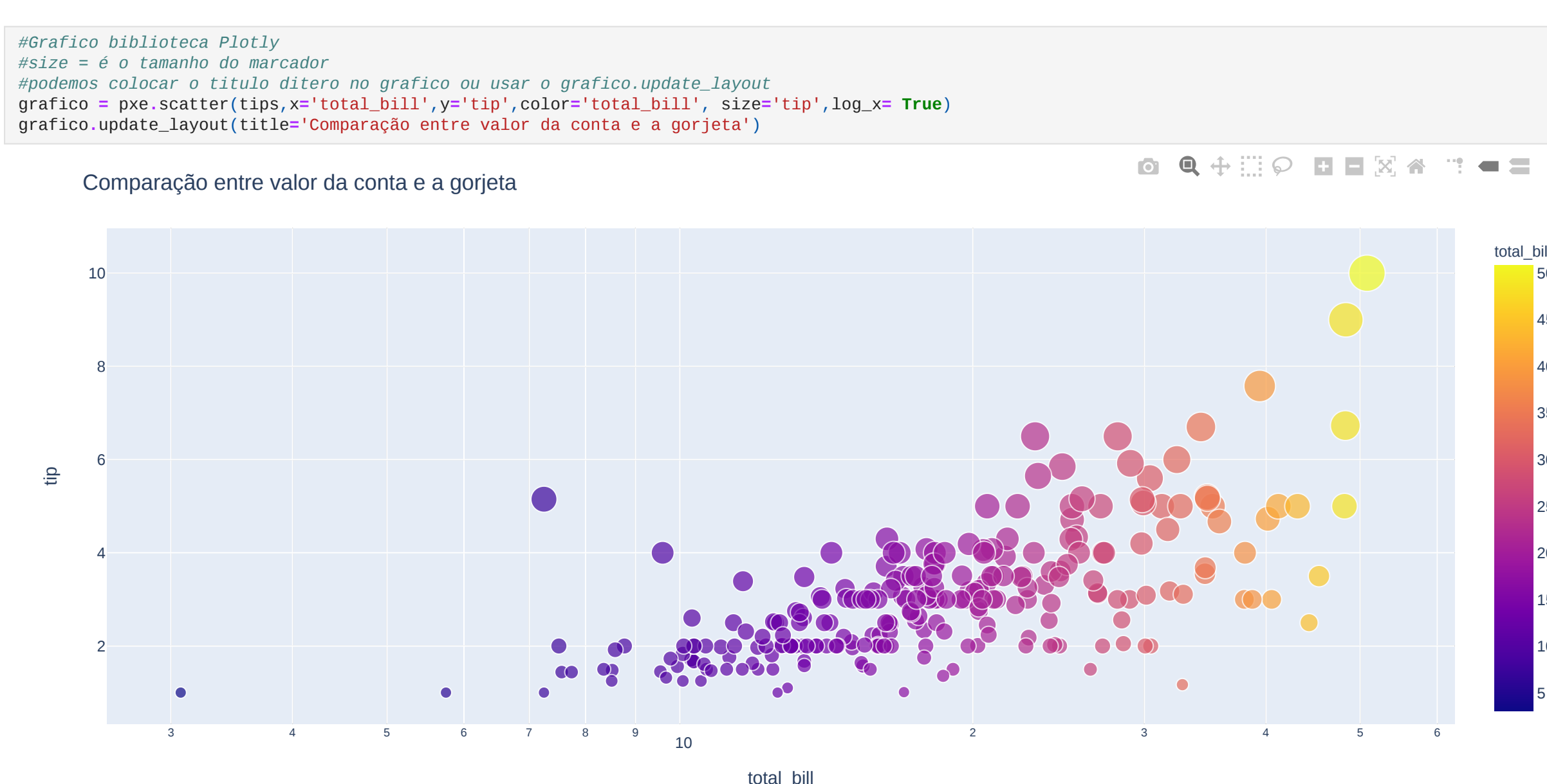
```
In [33]: #Grafico biblioteca Plotly
#size = é o tamanho do marcador
#size = é o tamanho do marcador
grafico = pxe.scatter(tips,x='total_bill',y='tip',color='sex', size='tip')
grafico.show()
```



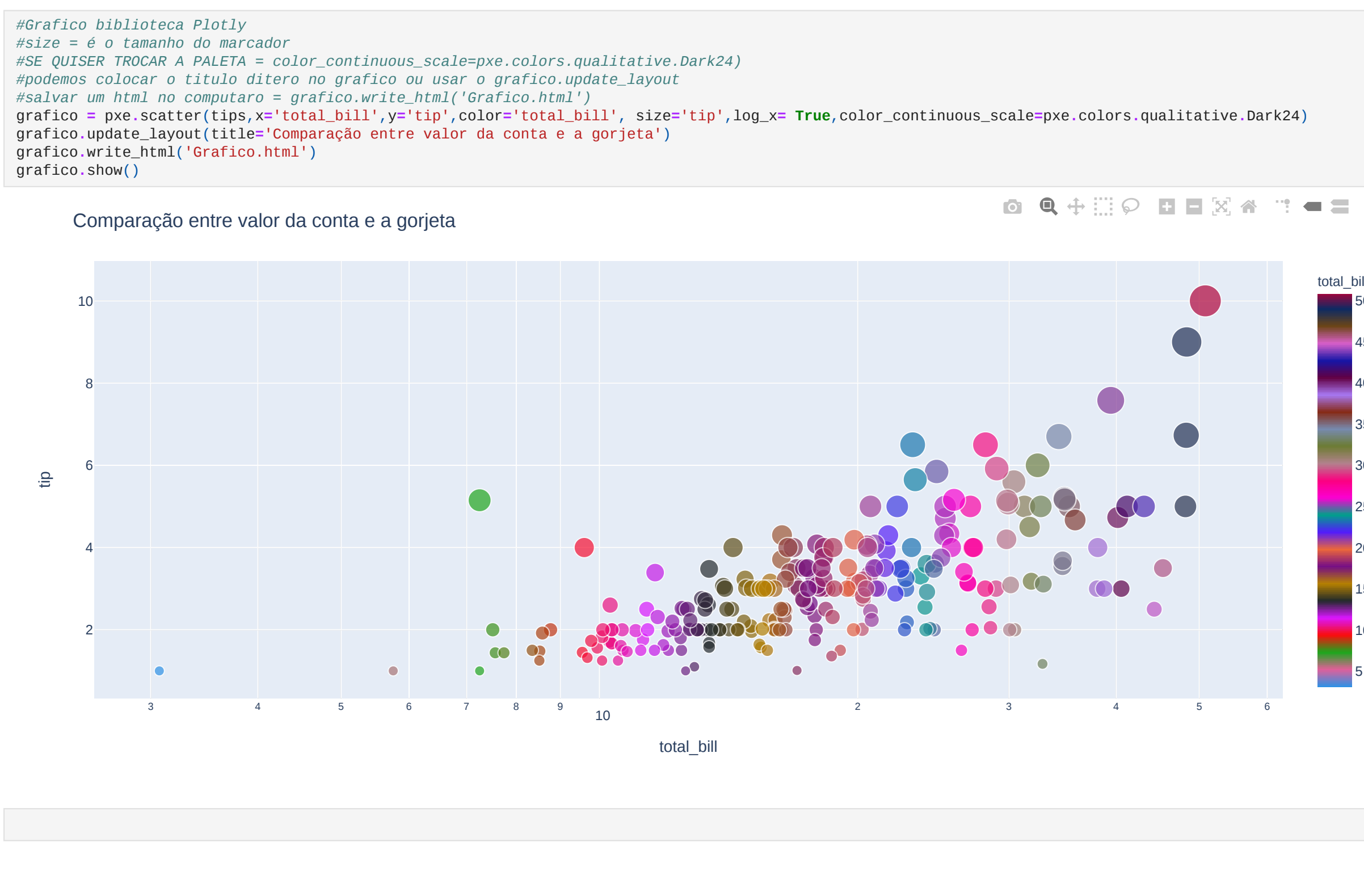
```
In [36]: #Grafico biblioteca Plotly
#size = é o tamanho do marcador
#size = é o tamanho do marcador
#podemos colocar o título dterno no grafico ou usar o grafico.update_layout
#podemos colocar o título dterno no grafico ou usar o grafico.update_layout
grafico = pxe.scatter(tips,x='total_bill',y='tip',color='sex', size='tip',log_x= True)
grafico.update_layout(title='Comparação entre valor da conta e a gorjeta')
```



```
In [38]: #Grafico biblioteca Plotly
#size = é o tamanho do marcador
#size = é o tamanho do marcador
#podemos colocar o título dterno no grafico ou usar o grafico.update_layout
#podemos colocar o título dterno no grafico ou usar o grafico.update_layout
grafico = pxe.scatter(tips,x='total_bill',y='tip',color='total_bill', size='tip',log_x= True)
grafico.update_layout(title='Comparação entre valor da conta e a gorjeta')
```



```
In [45]: #Grafico biblioteca Plotly
#size = é o tamanho do marcador
#size = é o tamanho do marcador
#SE QUISER TROCAR A PALETA = color_continuous_scale=pxe.colors.qualitative.Dark24)
#podemos colocar o título dterno no grafico ou usar o grafico.update_layout
#salvar um html no computador = grafico.write_html('Grafico.html')
grafico = pxe.scatter(tips,x='total_bill',y='tip',color='total_bill', size='tip',log_x= True,color_continuous_scale=pxe.colors.qualitative.Dark24)
grafico.write_html('Grafico.html')
grafico.show()
```



```
In [ ]:
```