In [3]:
```
import numpy as np
import pandas as pd
import matplotlib.pyplot as plt
%matplotlib inline
import seaborn as sns
sns.set_style('whitegrid')
sns.set_context('notebook')
import plotly.express as px
import warnings
warnings.filterwarnings("ignore")
```

In [5]:
```
from plotly.offline import download_plotlyjs, init_notebook_mode, plot, iplot
init_notebook_mode(connected = True)
import cufflinks as cf
cf.go_offline()
```

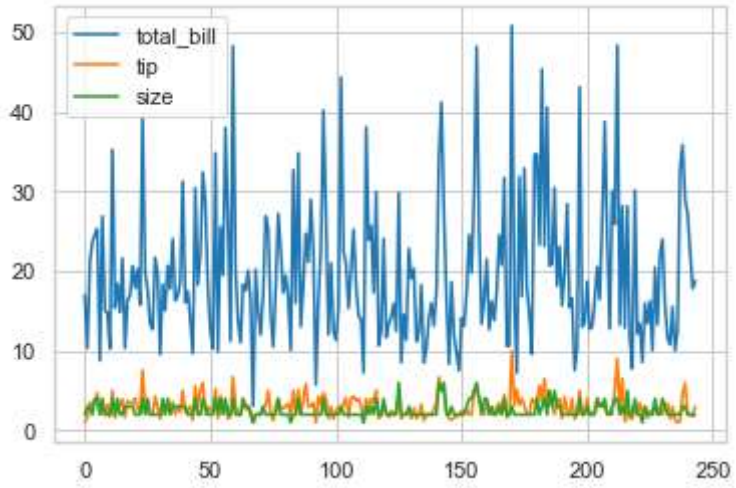
In [6]:
```
df = pd.read_csv(r"tips.csv")
df.head()
```

Out[6]:

	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 [7]:
```
df.plot()
```

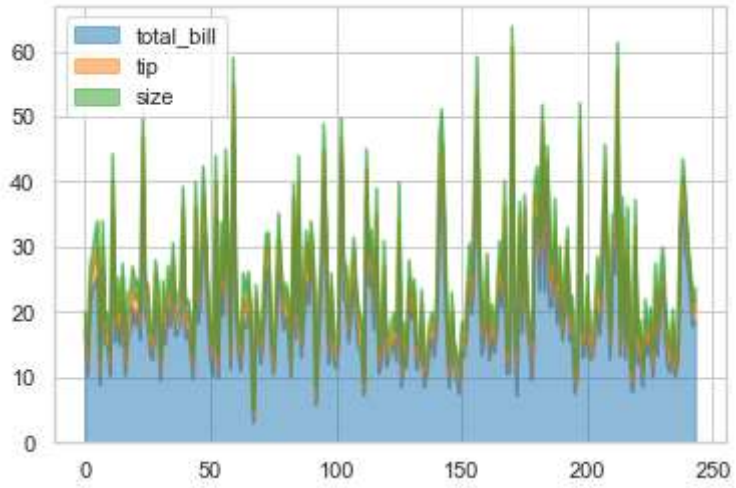
Out[7]:
```
<Axes: >
```



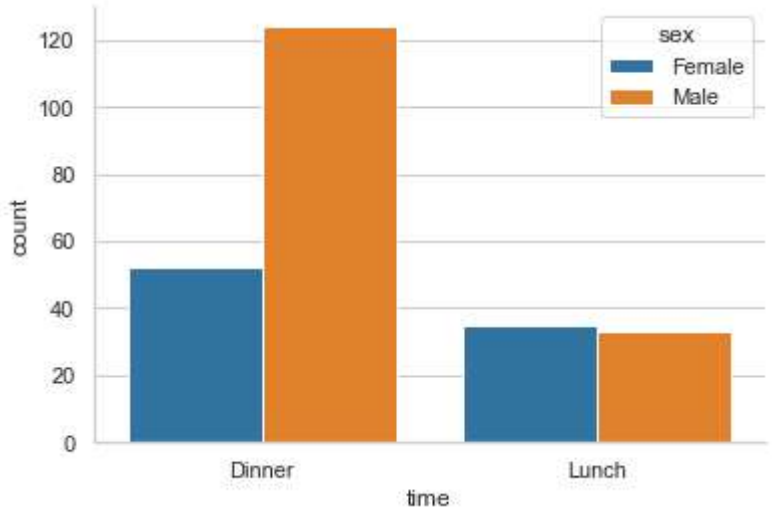
In [8]:
```
df.iplot()
```

In [9]:
```
df.plot.area(alpha=0.5)
```

Out[9]:
```
<Axes: >
```



In [12]:
```
sns.countplot(x=df.time, order = df.time.value_counts().index, hue = df.sex)
sns.despine()
```

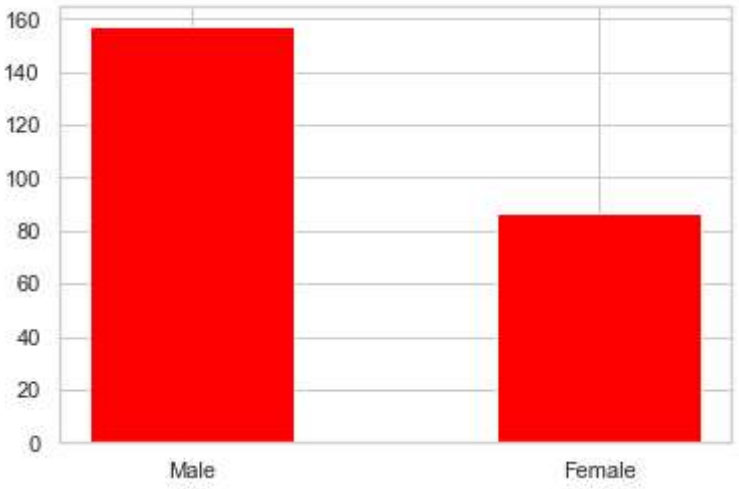


In [13]:
```
data = df.sex.value_counts()
data
```

Out[13]:
```
Male      157
Female     87
Name: sex, dtype: int64
```

In [14]:
```
plt.bar(data.index, data, width=0.5, color='red')
```

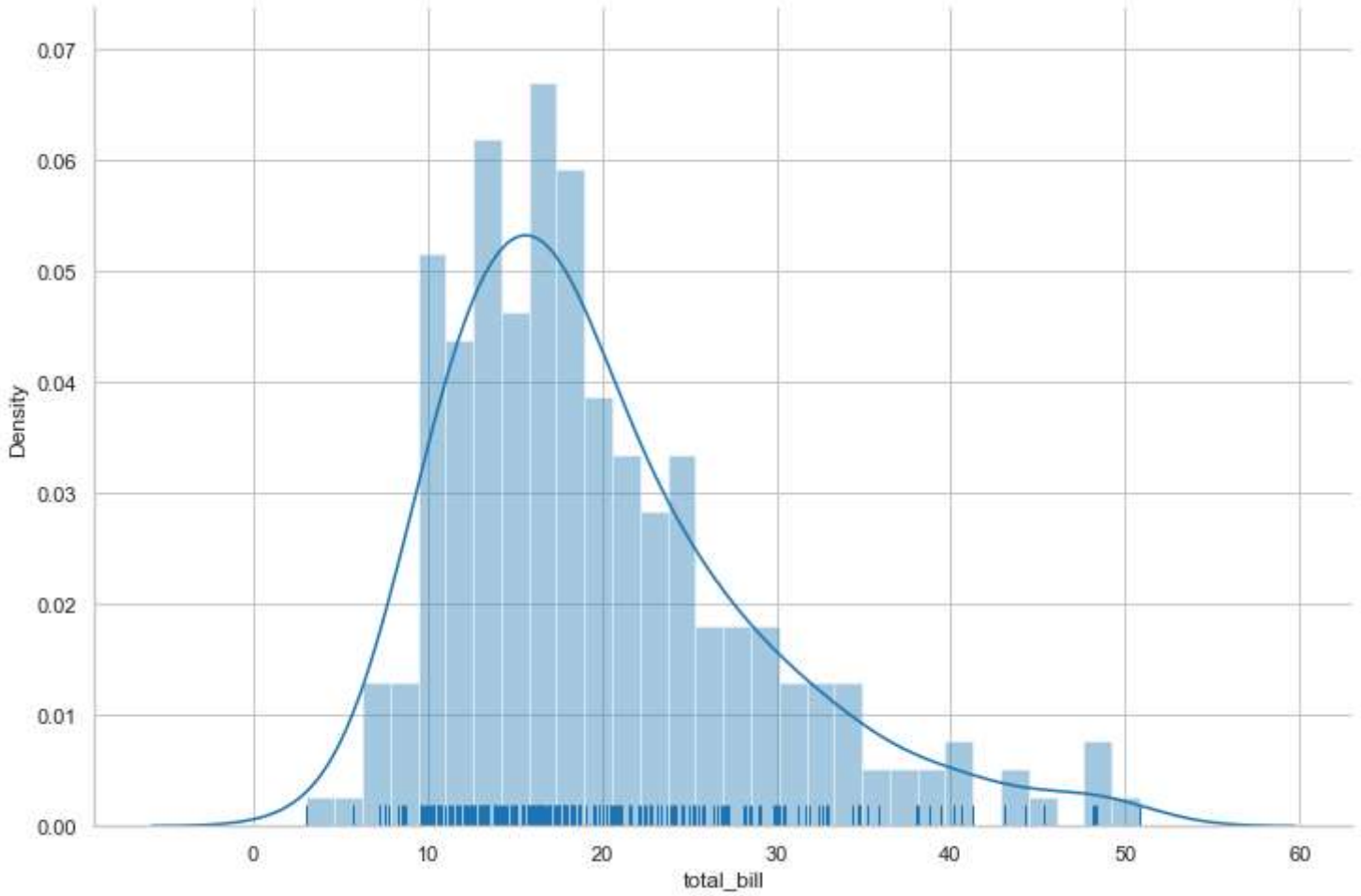
Out[14]: <BarContainer object of 2 artists>



In [15]:
```
fig = px.histogram(df.time, color= df.sex, barmode = 'group')
fig.update_layout(title_text = 'CountPlot', width=400, height = 300)
fig.update_xaxes(title_text = 'Time')
fig.update_yaxes(title_text = 'Count')
```

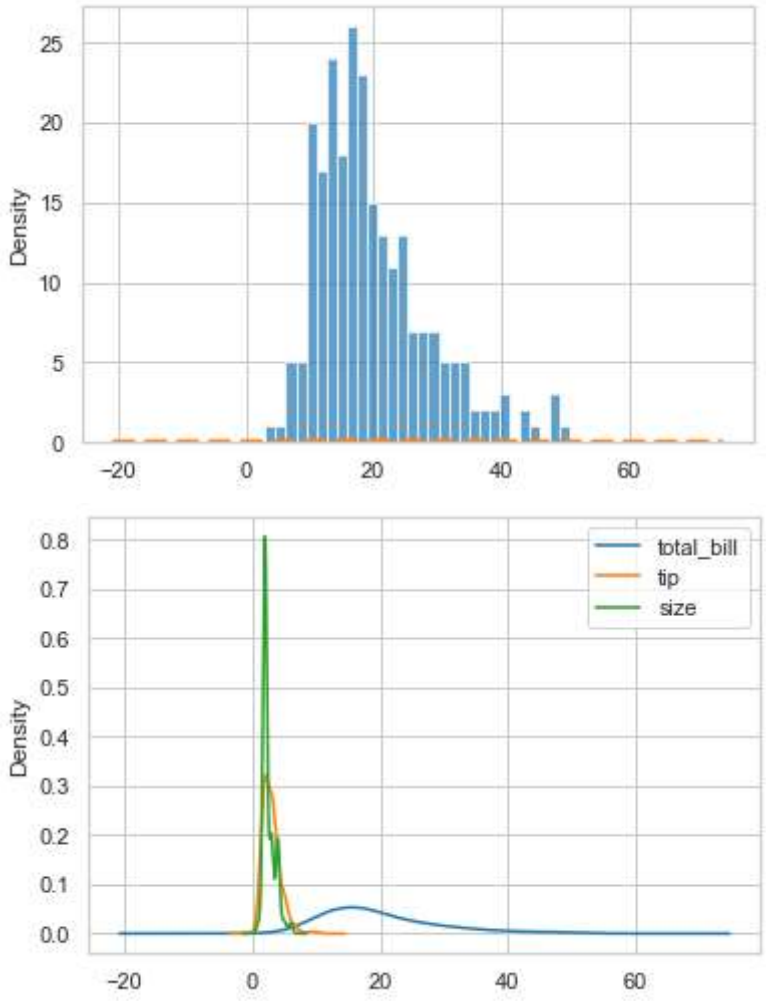
In [16]:
```
plt.figure(figsize=(12,8))
sns.distplot(df.total_bill, hist = True, kde =True,rug=True,bins=30)
```

```
sns.despine()
```



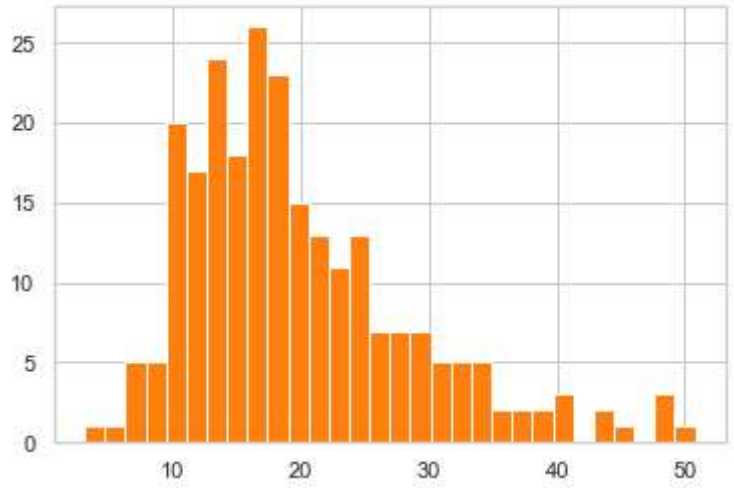
In [18]:
```
df.total_bill.hist(bins=30, alpha = 0.7)
df.total_bill.plot.kde(lw=3, ls='--')
df.plot.kde()
```

Out[18]:
```
<Axes: ylabel='Density'>
```



In [19]:
```
print(plt.hist(df['total_bill'], bins=30)[0])  #Counts
print(plt.hist(df['total_bill'], bins =30)[1]) #bins range
```

```
[ 1.  1.  5.  5. 20. 17. 24. 18. 26. 23. 15. 13. 11. 13.  7.  7.  7.  5.
  5.  5.  2.  2.  2.  3.  0.  2.  1.  0.  3.  1.]
[ 3.07        4.66133333  6.25266667  7.844       9.43533333 11.02666667
 12.618      14.20933333 15.80066667 17.392      18.98333333 20.57466667
 22.166      23.75733333 25.34866667 26.94       28.53133333 30.12266667
 31.714      33.30533333 34.89666667 36.488      38.07933333 39.67066667
 41.262      42.85333333 44.44466667 46.036      47.62733333 49.21866667
 50.81      ]
```



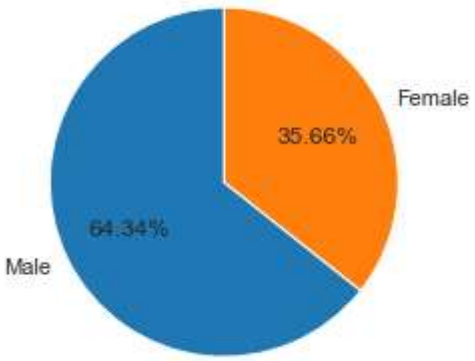
In [20]:
```
df.total_bill.iplot(kind='hist', bins=30)
```

In [21]:
```
fig = px.histogram(df.total_bill, marginal="box", title = "Year Count Plot")
fig.update_layout(width=800, height = 550)
fig.update_xaxes(title_text = 'Total_bill')
fig.update_yaxes(title_text = 'Count')
```

In [22]:
```
data = df.sex.value_counts()
data
```

Out[22]:
```
Male      157
Female     87
Name: sex, dtype: int64
```

In [23]:
```
plt.pie(data, labels= data.index, startangle = 90, autopct = '%1.2f%%')
plt.show()
```



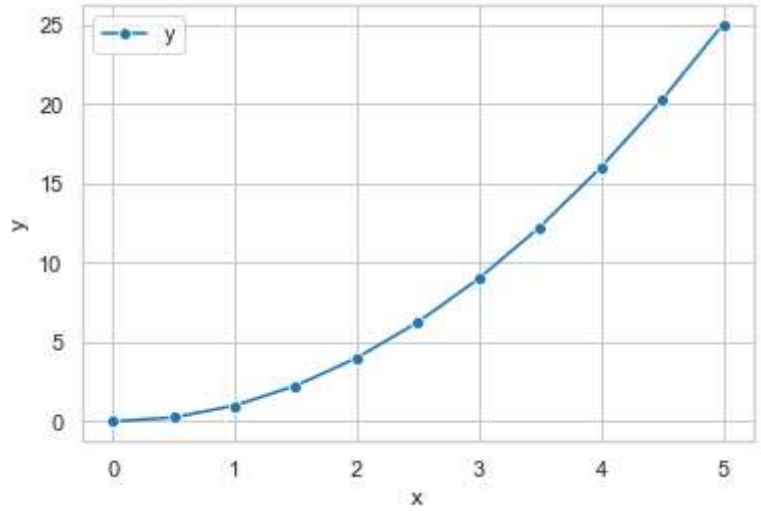
In [24]:
```
fig = px.pie(values=data, names=data.index)
fig.update_layout(title_text = 'Gender Distribution', width=500, height = 400)
```

In [25]:
```
df = pd.DataFrame({'x':np.linspace(0,5,11), 'y': np.linspace(0,5,11)**2})
df.head()
```

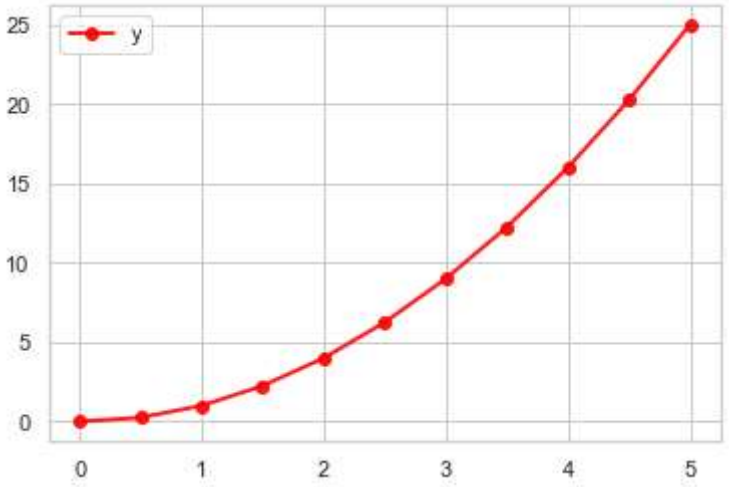
Out[25]:

	x	y
0	0.0	0.00
1	0.5	0.25
2	1.0	1.00
3	1.5	2.25
4	2.0	4.00

In [28]:
```
sns.lineplot(x='x', y='y', data=df, marker='o', label='y')
plt.show()
```



In [31]:
```
plt.plot(df['x'], df['y'], 'r-o', lw=2, alpha=0.9, label='y')
plt.legend()
plt.show()
```



Jointplot : {scatter,hex,reg}

In [34]:
```
import numpy as np
import pandas as pd
import matplotlib.pyplot as plt
%matplotlib inline
import seaborn as sns
sns.set_style('whitegrid')
sns.set_context('notebook')
import plotly.express as px
import warnings
warnings.filterwarnings("ignore")
```

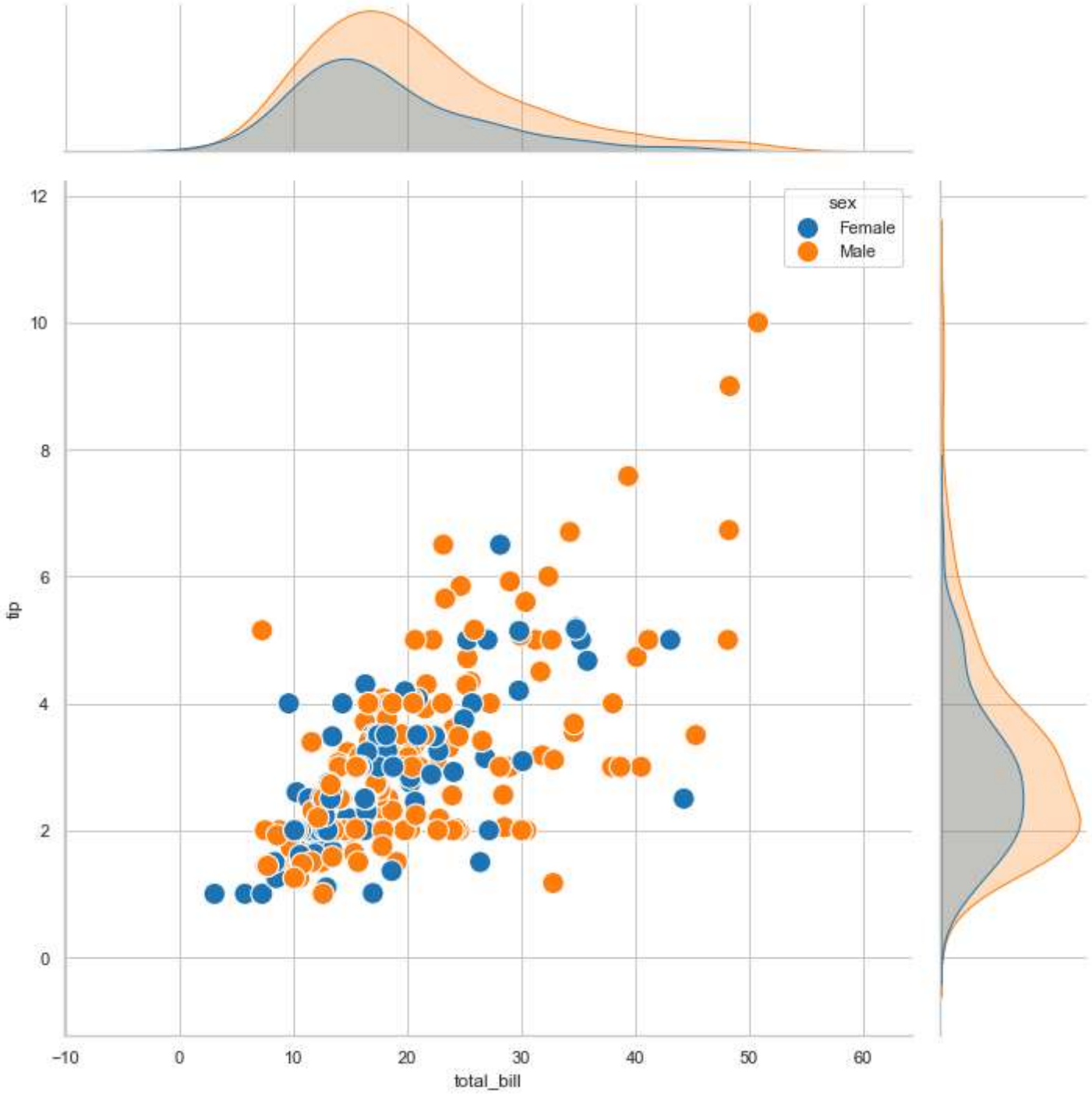
In [36]:
```
df = pd.read_csv(r"tips.csv")
df.head()
```

Out[36]:

	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

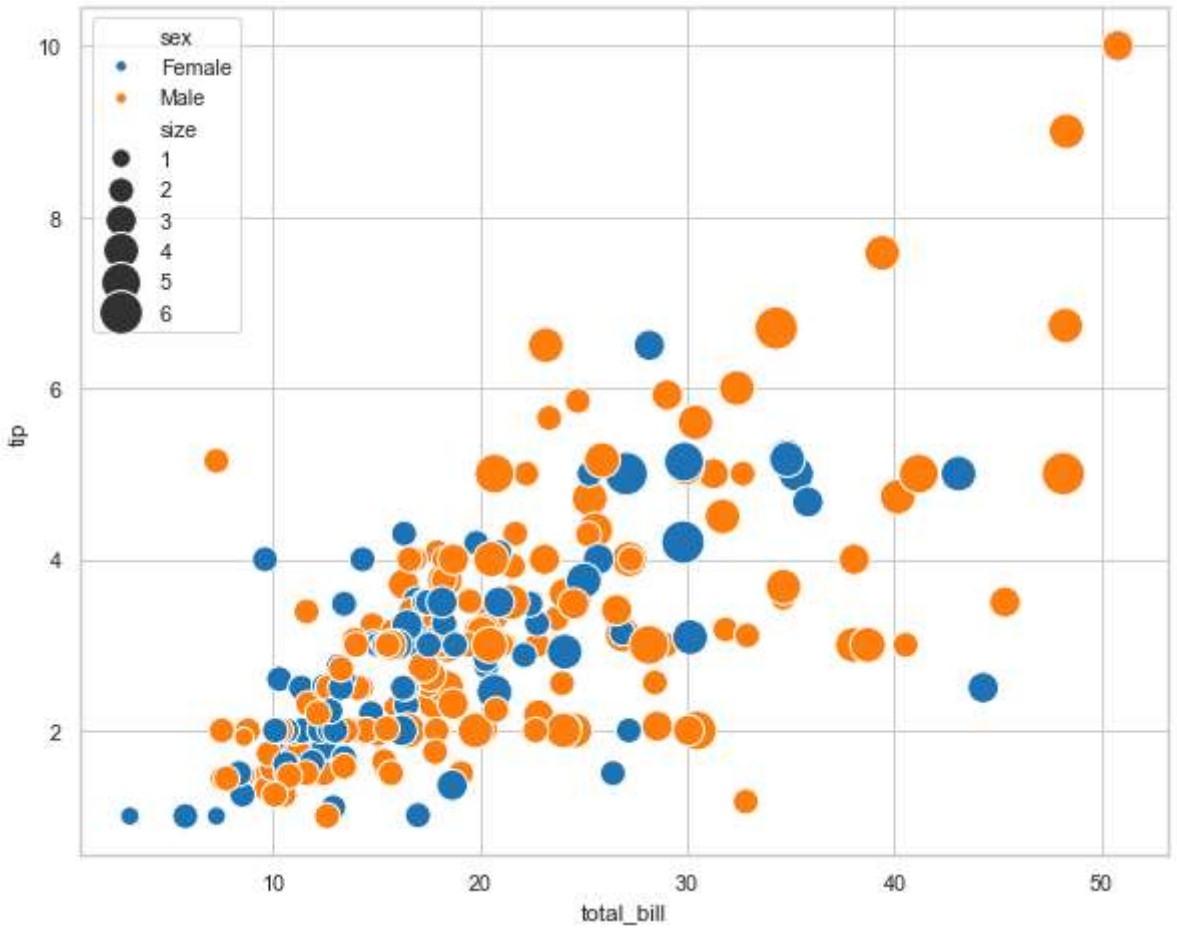
In [38]:
```
sns.jointplot(x='total_bill', y='tip', data=df, kind='scatter', hue='sex', height=10, ratio=5, joint_kws={'s': 200})
```

Out[38]:
```
<seaborn.axisgrid.JointGrid at 0x1de2e412e20>
```


sex
Female
Male
tip
total_bill
12
10
8
6
4
2
0
−10
0
10
20
30
40
50
60

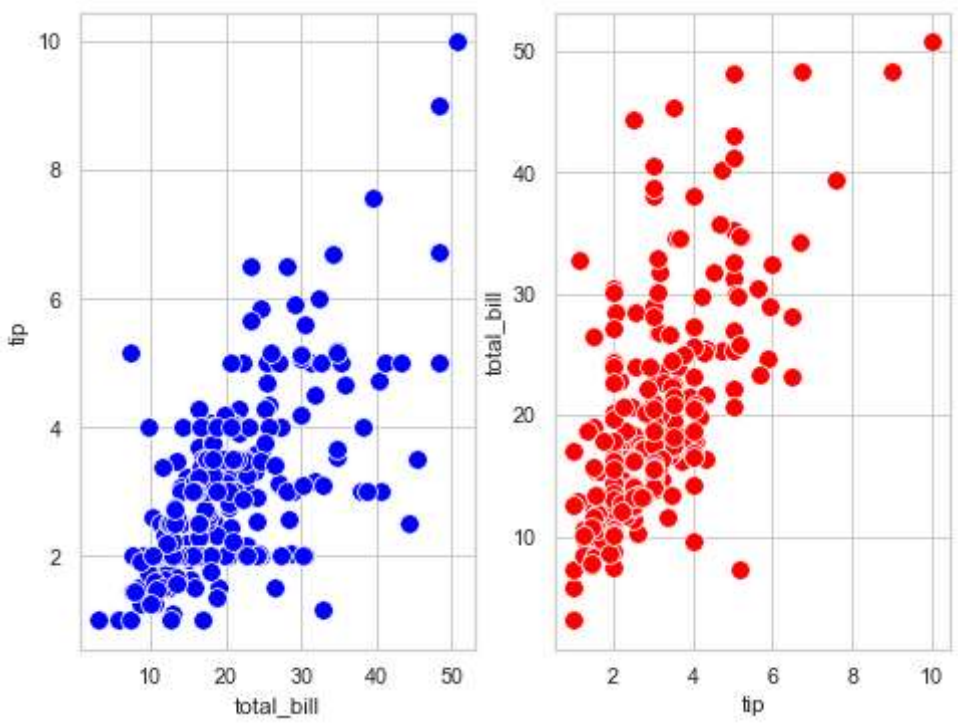
In [46]:
```
plt.figure(figsize=(10,8),dpi =70)
sns.scatterplot(x='total_bill', y='tip', data=df, color='b', hue='sex', size='size', sizes=(100, 500))
```

Out[46]:
```
<Axes: xlabel='total_bill', ylabel='tip'>
```



In [48]:
```
plt.figure(figsize = (8,6), dpi=70)
plt.subplot(1,2,1)
sns.scatterplot(x='total_bill', y='tip', data=df, s=100, color='b')
plt.subplot(1,2,2)
sns.scatterplot(x='tip', y='total_bill',data = df, s= 100, color ='r')
```

Out[48]:
```
<Axes: xlabel='tip', ylabel='total_bill'>
```

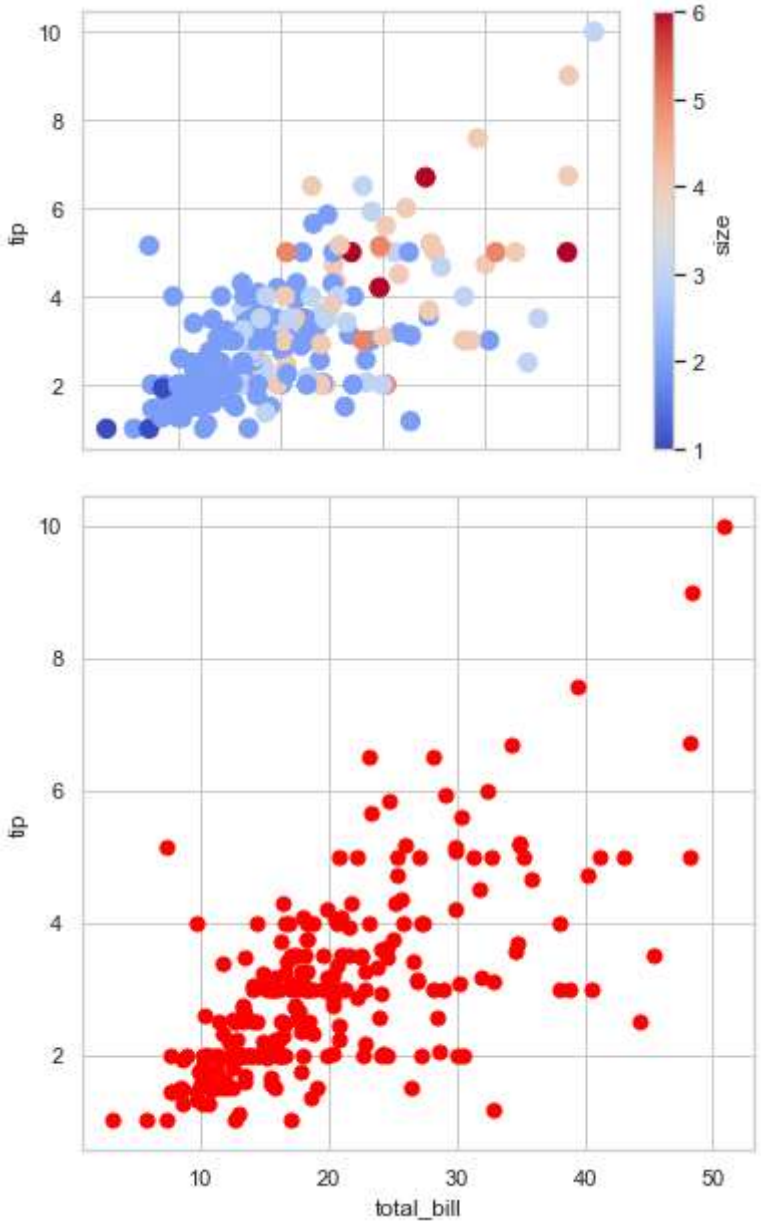


In [49]:
```
df.iplot(kind='scatter', x= 'total_bill', y='tip', mode = 'markers', size=10)
```

In [50]:
```
fig = px.scatter(x=df['total_bill'], y=df['tip'], title="Total bill vs Tip", color=df['sex'], size_max=20)

fig.update_layout(width=1200, height=500)
fig.update_xaxes(title_text='Total_bill')
fig.update_yaxes(title_text='Tip')

fig.show()
```

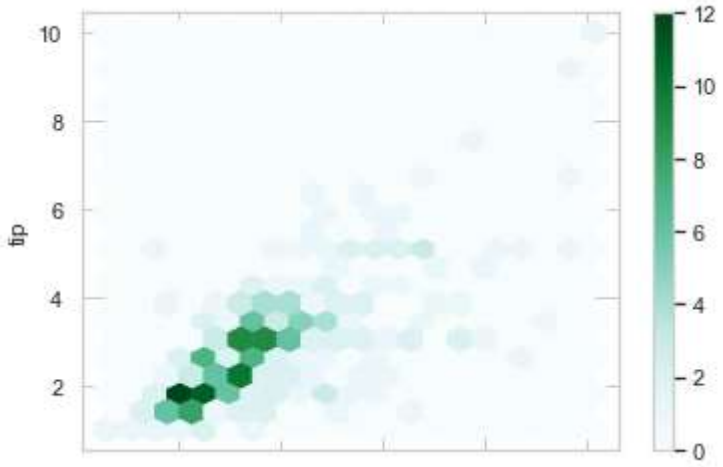
In [51]:
```
plt.figure(figsize=(8,6), dpi=70)
df.plot.scatter(x='total_bill',y='tip', c= 'size',s = df.size/20, cmap='coolwarm')
df.plot.scatter(x = 'total_bill',y = 'tip', c='red',s =50, figsize=(6,6))
```

Out[51]:
```
<Axes: xlabel='total_bill', ylabel='tip'>

<Figure size 560x420 with 0 Axes>
```



In [52]:
```
df.plot.hexbin(x='total_bill', y='tip', gridsize=20)
```

Out[52]:
```
<Axes: xlabel='total_bill', ylabel='tip'>
```

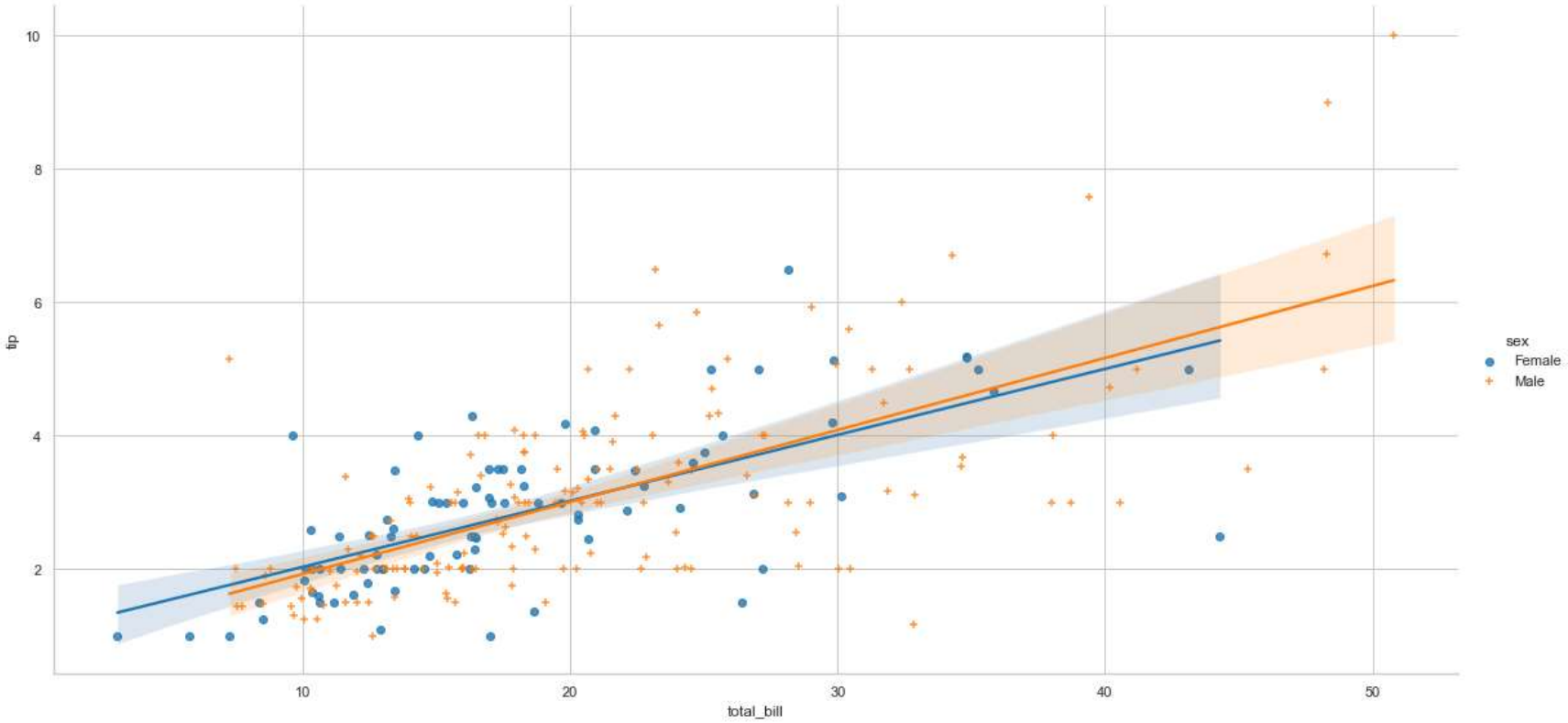


In [53]:
```
df.iplot(kind = 'bubble',x='total_bill',y='tip', size='size')
```

Regression Plot

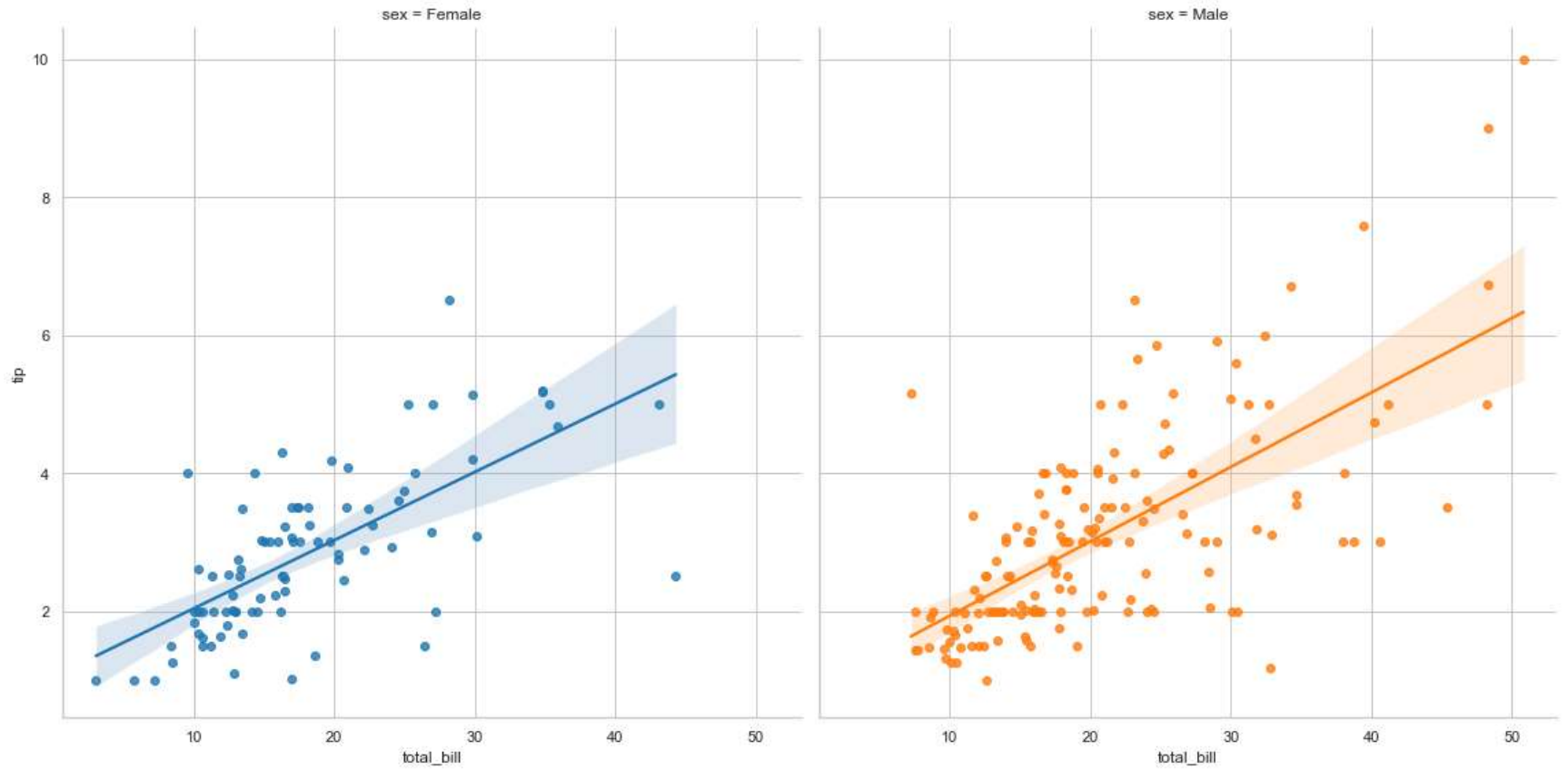
In [54]:
```
sns.lmplot(data = df, x = 'total_bill', y= 'tip', hue='sex',markers = ['o','+'],height=8, aspect = 2)
```

Out[54]:
```
<seaborn.axisgrid.FacetGrid at 0x1de3101bbb0>
```



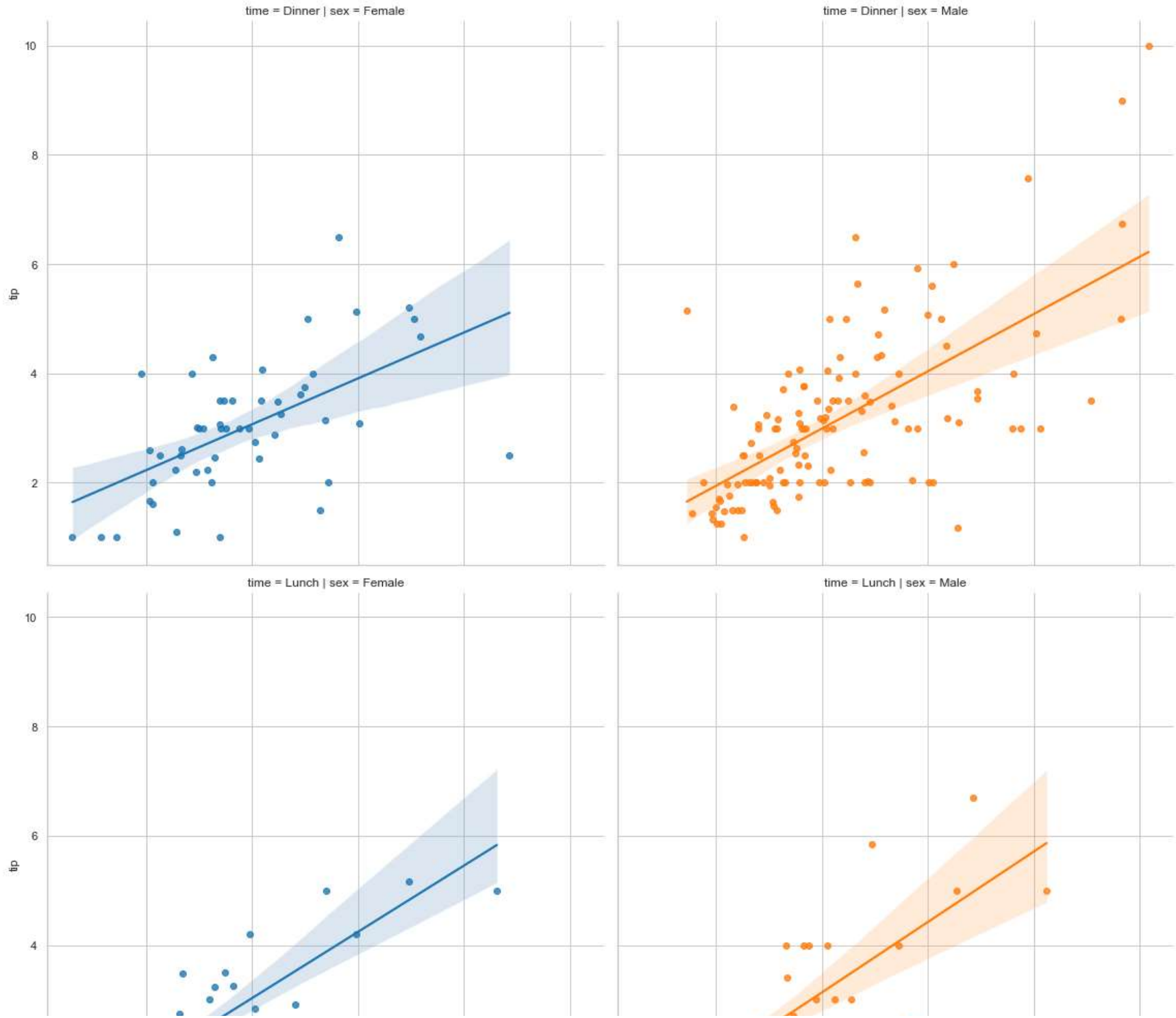
In [55]:
```
sns.lmplot(data=df, x='total_bill', y='tip', hue='sex',col='sex', height=8)
```

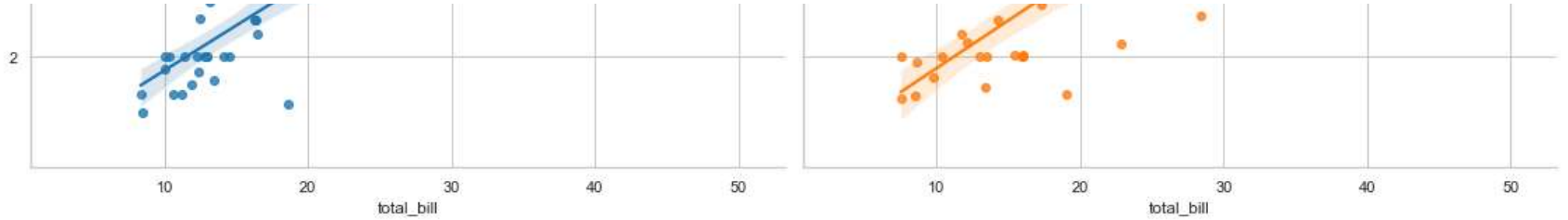
Out[55]:
```
<seaborn.axisgrid.FacetGrid at 0x1de3102f310>
```



In [57]:
```
sns.lmplot(data=df, x ='total_bill', y='tip', hue='sex', col='sex',row='time',height=8)
```

Out[57]:
```
<seaborn.axisgrid.FacetGrid at 0x1de3101b8b0>
```

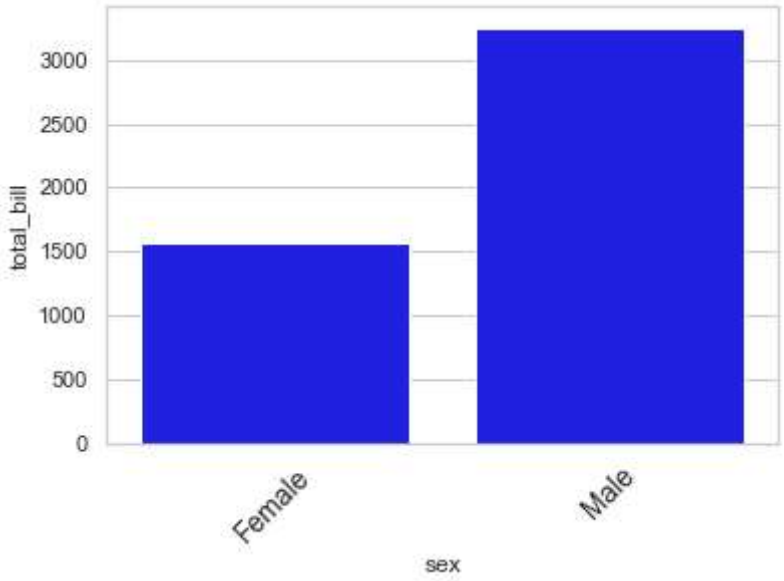

time = Dinner | sex = Female
time = Dinner | sex = Male
time = Lunch | sex = Female
time = Lunch | sex = Male
tip
10
8
6
4
2



Categorical vs Numerical

In [61]:
```
sns.barplot(x='sex', y='total_bill', data=df, estimator=np.sum, ci=False, color='blue')
plt.xticks(rotation=45, fontsize=14)
```

Out[61]:
```
([0, 1], [Text(0, 0, 'Female'), Text(1, 0, 'Male')])
```

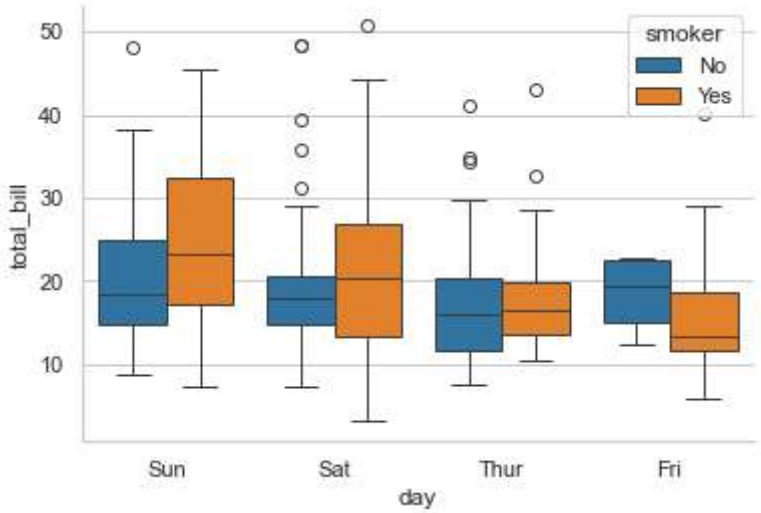


In [62]:
```
df.iplot(kind ='bar', x = 'sex', y='total_bill')
```

In [63]:
```
df.mean().iplot(kind='bar')
```

Box & Wiskers plot (Describe plot)

In [64]:
```
sns.boxplot(x='day',y='total_bill',data=df, hue='smoker')
sns.despine()
```



In [68]:
```
fig = px.box(x=df.day, y=df.total_bill, color=df.smoker)
fig.update_layout(title_text = 'BoxPlot',height = 500)
fig.update_xaxes(title_text='Day')
fig.update_yaxes(title_text='Total Bill')
```

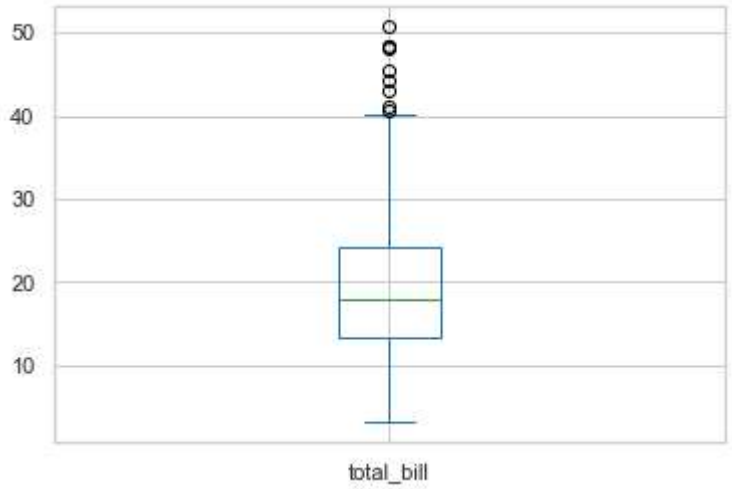
In [69]:
```
pd.DataFrame(df.describe()).T
```

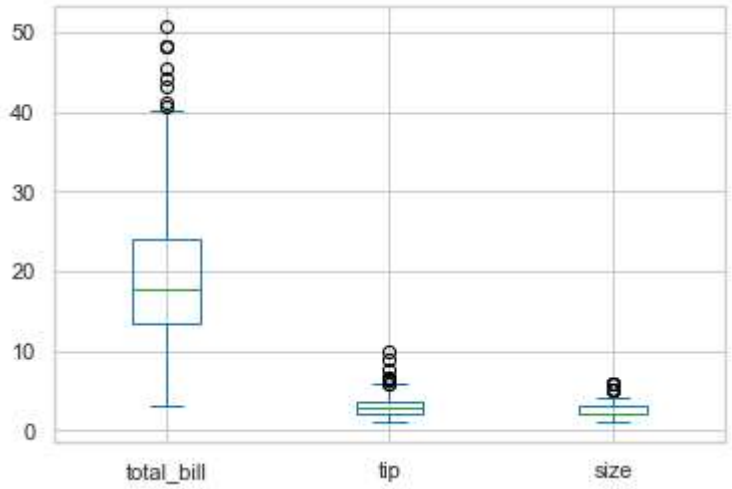
Out[69]:

	count	mean	std	min	25%	50%	75%	max
total_bill	244.0	19.785943	8.902412	3.07	13.3475	17.795	24.1275	50.81
tip	244.0	2.998279	1.383638	1.00	2.0000	2.900	3.5625	10.00
size	244.0	2.569672	0.951100	1.00	2.0000	2.000	3.0000	6.00

In [70]:
```
df.total_bill.plot.box()
df.plot.box()
```

Out[70]: <Axes: >



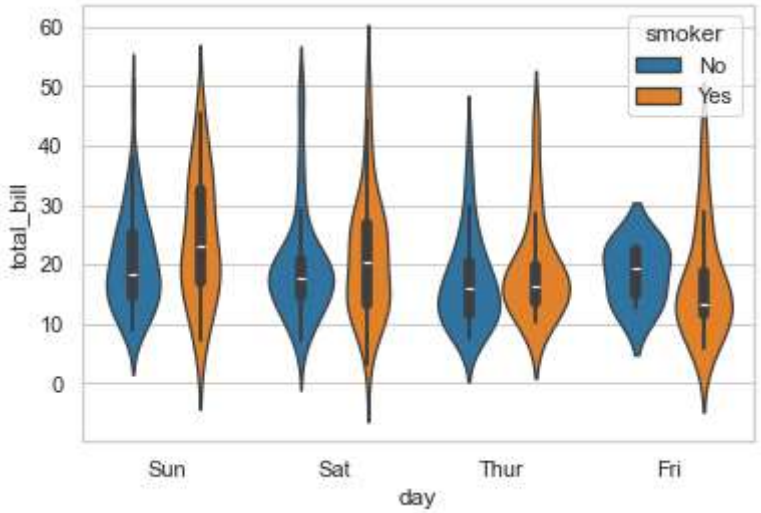


In [71]:
```
df.iplot(kind='box')
```

In [ ]:
```
# Violine Plot + Swarm Plot
```

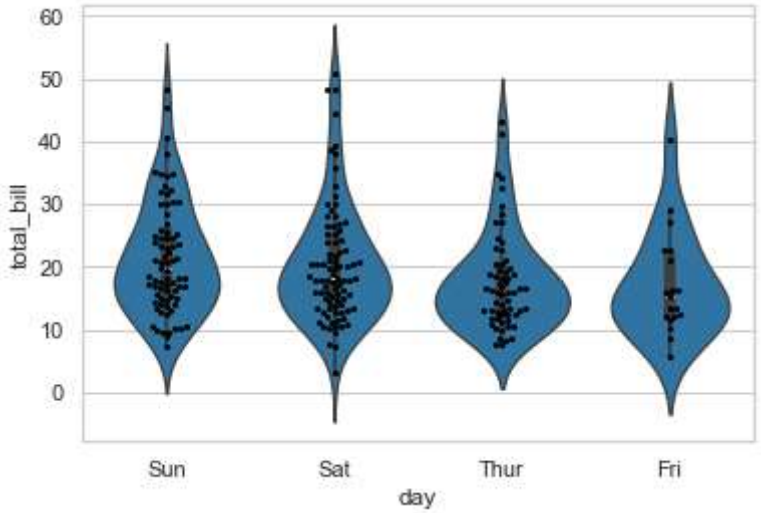
In [73]:
```
sns.violinplot(x='day', y='total_bill', hue='smoker', data=df, split=False)
```

Out[73]:
```
<Axes: xlabel='day', ylabel='total_bill'>
```



In [77]:
```
sns.violinplot(x=df.day,y=df.total_bill)
sns.swarmplot(x= df.day, y = df.total_bill,color='black',size = 3)
```

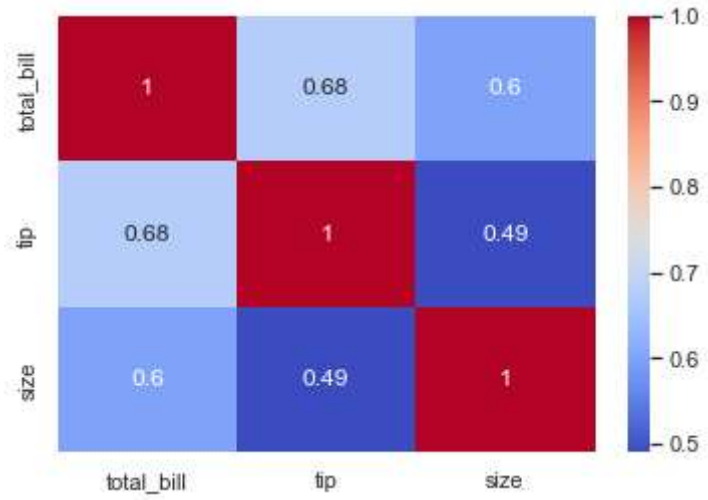
Out[77]:
```
<Axes: xlabel='day', ylabel='total_bill'>
```



In [78]:
```
# Heatmap
```

In [79]:
```
sns.heatmap(df.corr(), annot=True, cmap= 'coolwarm')
```

Out[79]:
```
<Axes: >
```



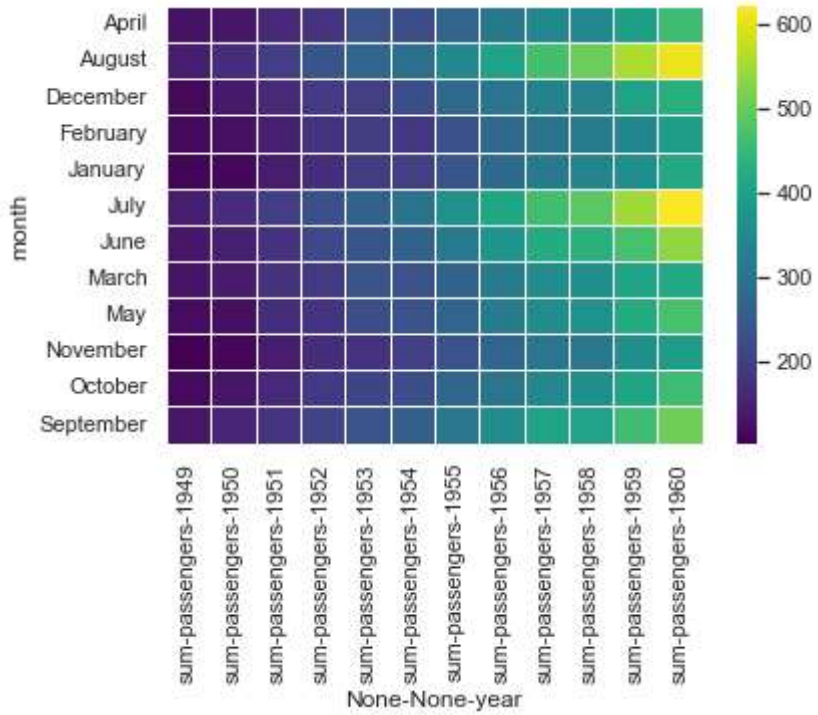
In [80]:
```
df2 = pd.read_csv(r"flights.csv")
df2.head(3)
```

Out[80]:

	year	month	passengers
0	1949	January	112
1	1949	February	118
2	1949	March	132

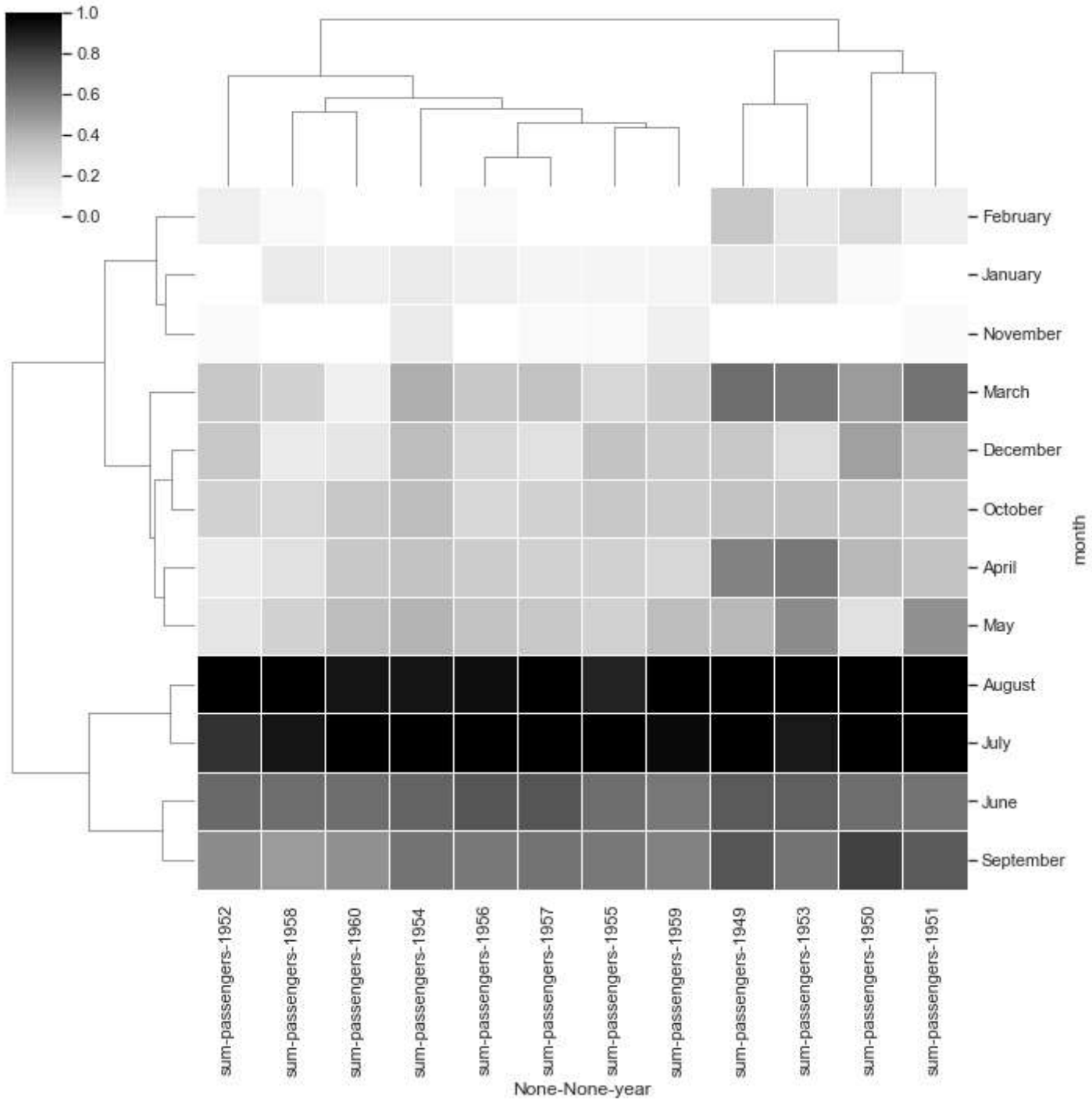
In [83]:
```
row=['month']
col = ['year']
value = ['passengers']
aggfun = ['sum']
sns.heatmap(df2.pivot_table(value, row, col, aggfun), annot = False, cmap = 'viridis', lw=1, linecolor = 'white')
```

Out[83]:
```
<Axes: xlabel='None-None-year', ylabel='month'>
```



In [84]:
```
row=['month']
col = ['year']
value = ['passengers']
aggfun = ['sum']
sns.clustermap(df2.pivot_table(value, row, col, aggfun), annot = False, cmap = 'Greys', lw=1, linecolor = 'white',stand
```

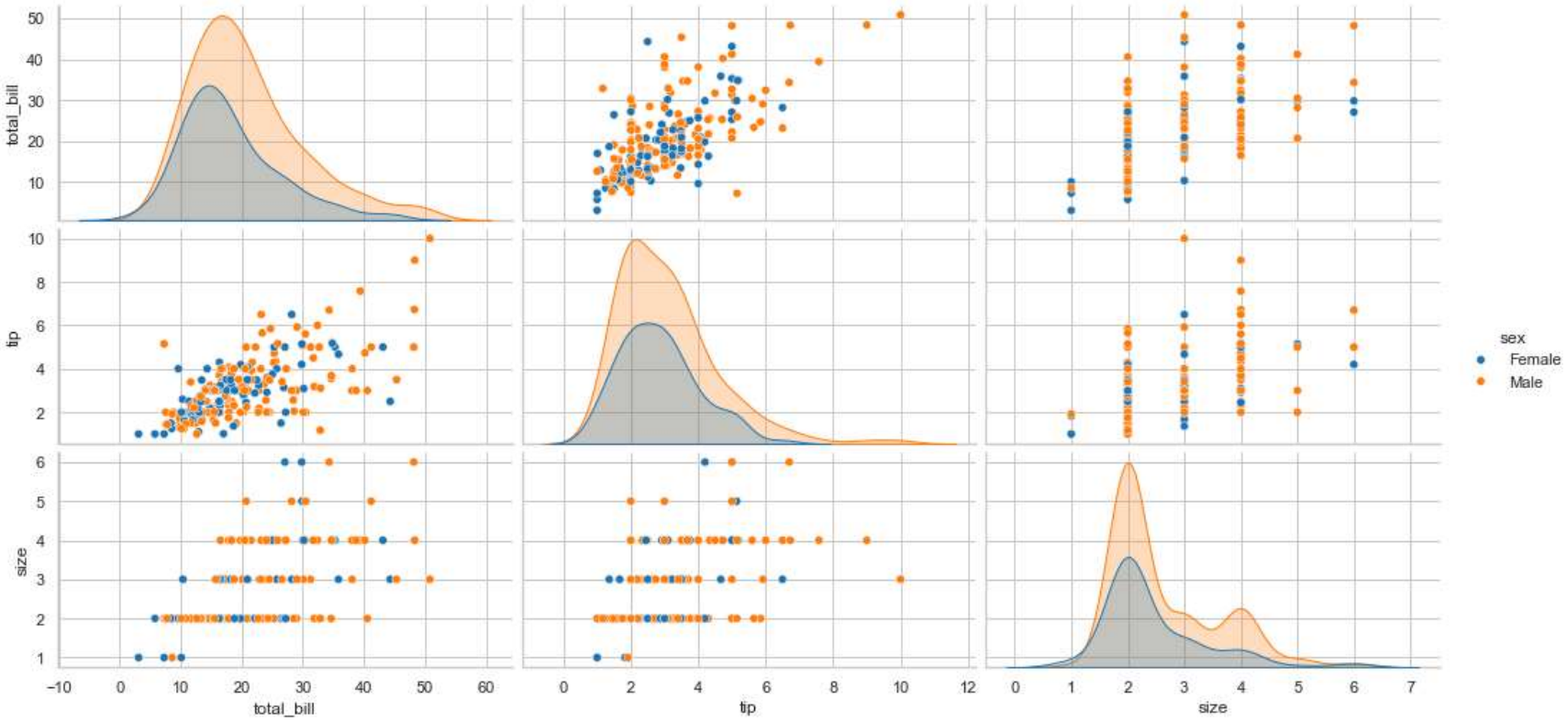
Out[84]:
```
<seaborn.matrix.ClusterGrid at 0x1de34bbf9a0>
```


1.0
0.8
0.6
0.4
0.2
0.0
February
January
November
March
December
October
April
May
August
July
June
September
month
sum-passengers-1952
sum-passengers-1958
sum-passengers-1960
sum-passengers-1954
sum-passengers-1956
sum-passengers-1957
sum-passengers-1955
sum-passengers-1959
sum-passengers-1949
sum-passengers-1953
sum-passengers-1950
sum-passengers-1951
None-None-year

In [85]:
```
#Pairplot (DatFrame plot)
```

In [87]:
```
sns.pairplot(df, hue='sex', height=2.4, aspect=2)
```

Out[87]:
```
<seaborn.axisgrid.PairGrid at 0x1de38956ee0>
```



In [88]:
```
df[['total_bill', 'tip', 'size']].scatter_matrix(size=3)
```