Visualization using seaborn

[1]:
```
import numpy as np
import pandas as pd
import matplotlib.pyplot as plt
import seaborn as sns
```

[2]:
```
data1 = pd.read_csv("datasets/iris.csv")
data2 = pd.read_csv("datasets/tips.csv")
```

[3]:
```
data1.head()
```

[3]:

	Sepal Length	Sepal Width	Petal Length	Petal Width	iris	species	no
0	5.1	3.5	1.4	0.2	Iris-setosa		1
1	4.9	3.0	1.4	0.2	Iris-setosa		1
2	4.7	3.2	1.3	0.2	Iris-setosa		1
3	4.6	3.1	1.5	0.2	Iris-setosa		1
4	5.0	3.6	1.4	0.2	Iris-setosa		1

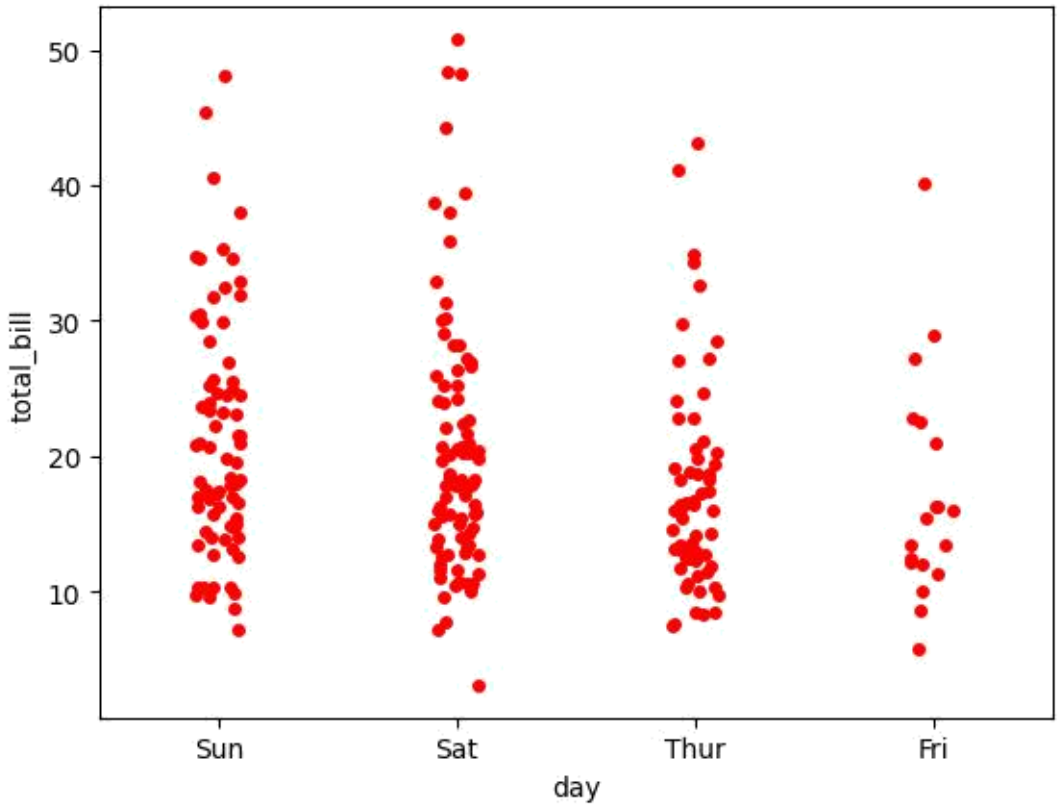
[4]:
```
data2.head()
```

[4]:

	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

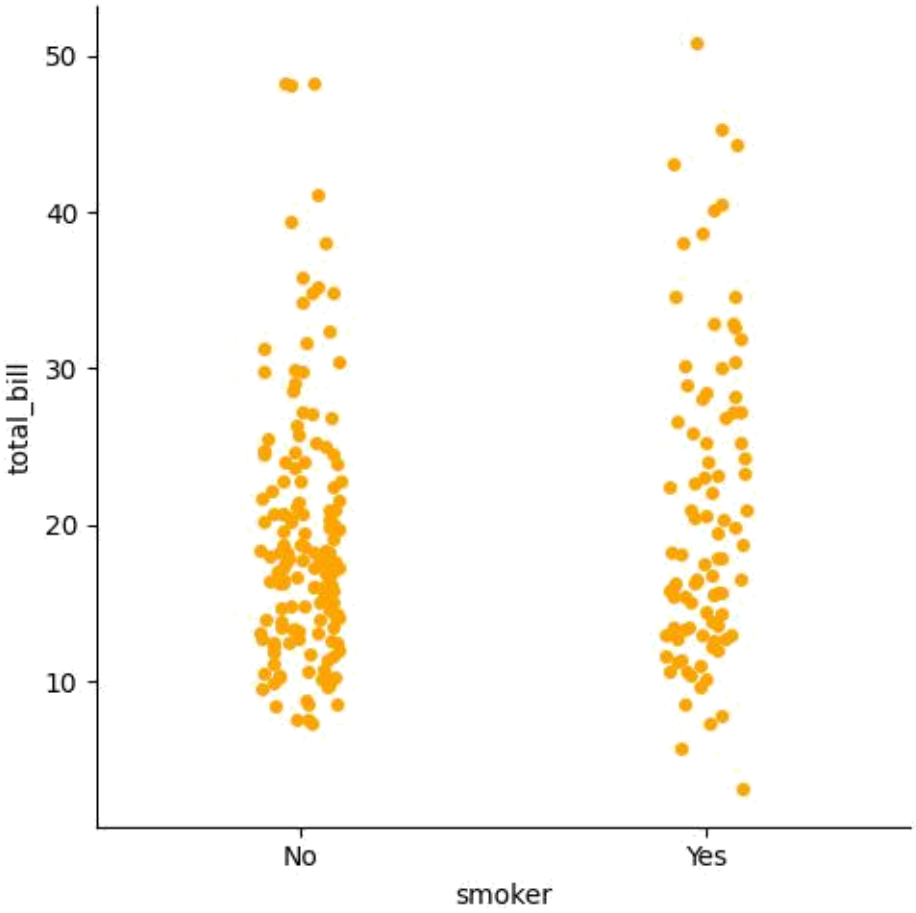
[5]:
```
# 1. swarm plot 2. strip plot |categorical scatter plot
```

[6]:
```
sns.stripplot(data= data2,x="day",y="total_bill",color="red",size=5)
plt.show()
```



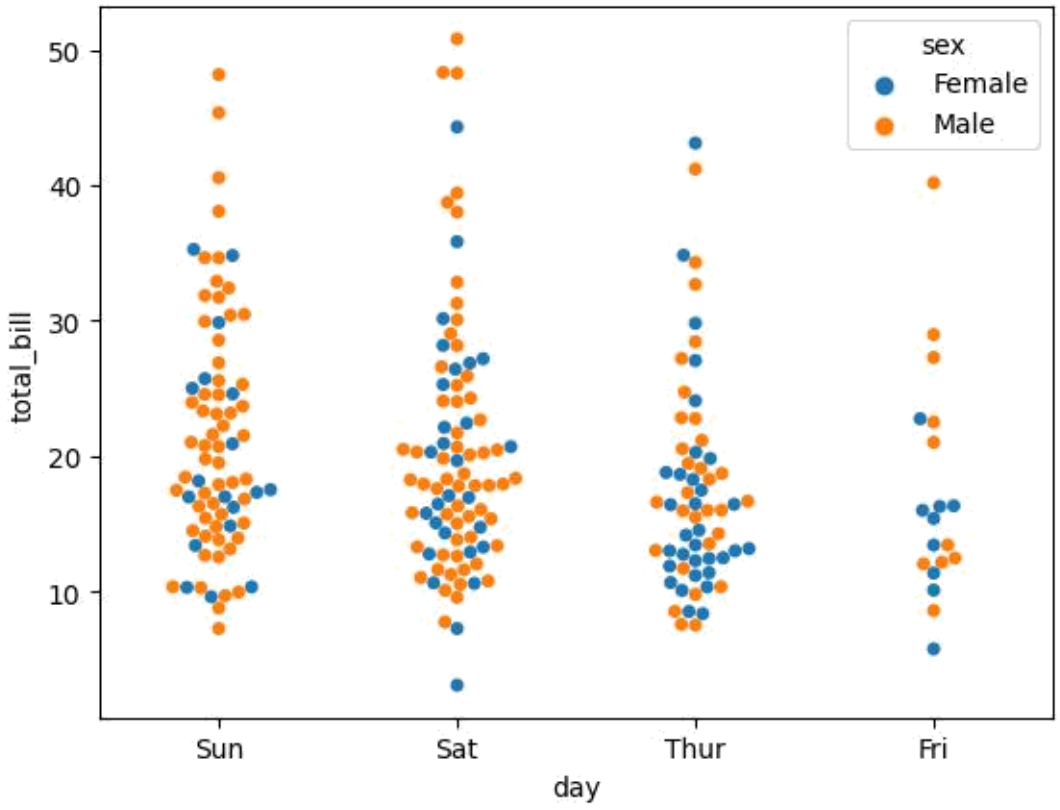
[7]:
```
sns.catplot(data=data2,kind="strip",x="smoker",y="total_bill",color="orange",size=5)
plt.show()
```

```
C:\Users\Admin\AppData\Local\Programs\Python\Python311\Lib\site-packages\seaborn\axisgrid.py:118: UserWarning: The figure layout has changed to tight
  self._figure.tight_layout(*args, **kwargs)
```



[8]:
```
  # swarm plot
# swarmplot is same as strip plot but it also shows
# the distriution of the data that how data is distubeted .
```

[9]:
```
sns.swarmplot(data=data2,x="day",y="total_bill",hue="sex")
plt.show()
```



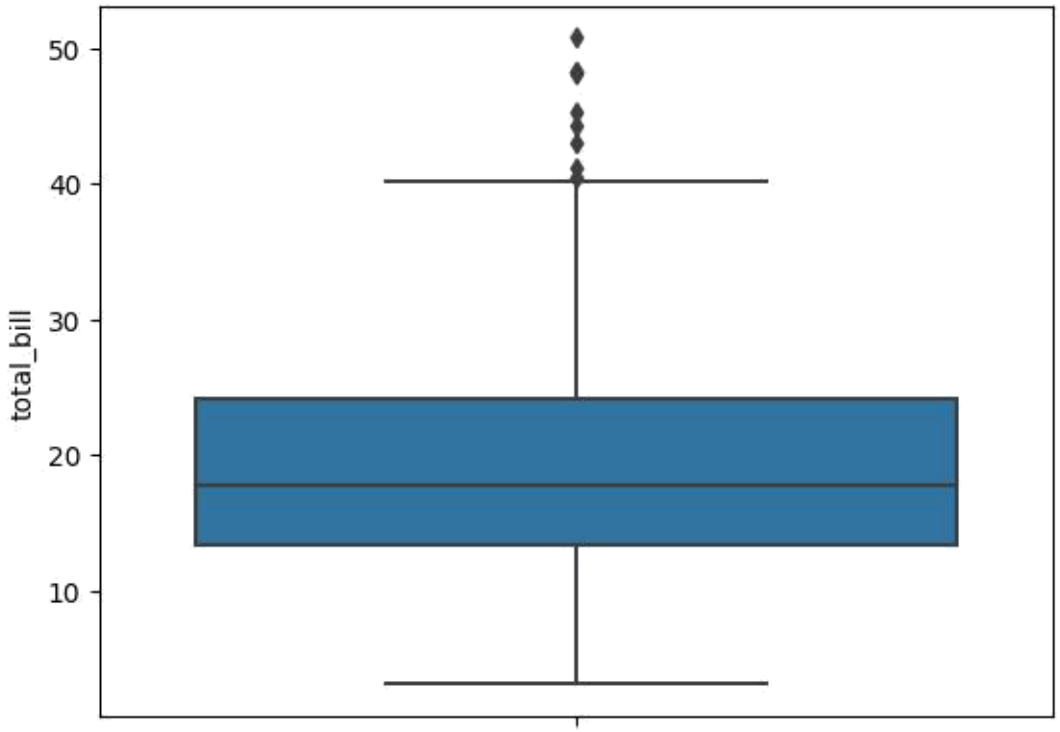
[10]:
```
# categorical distribution plot
```

[11]:
```
# box plot
```

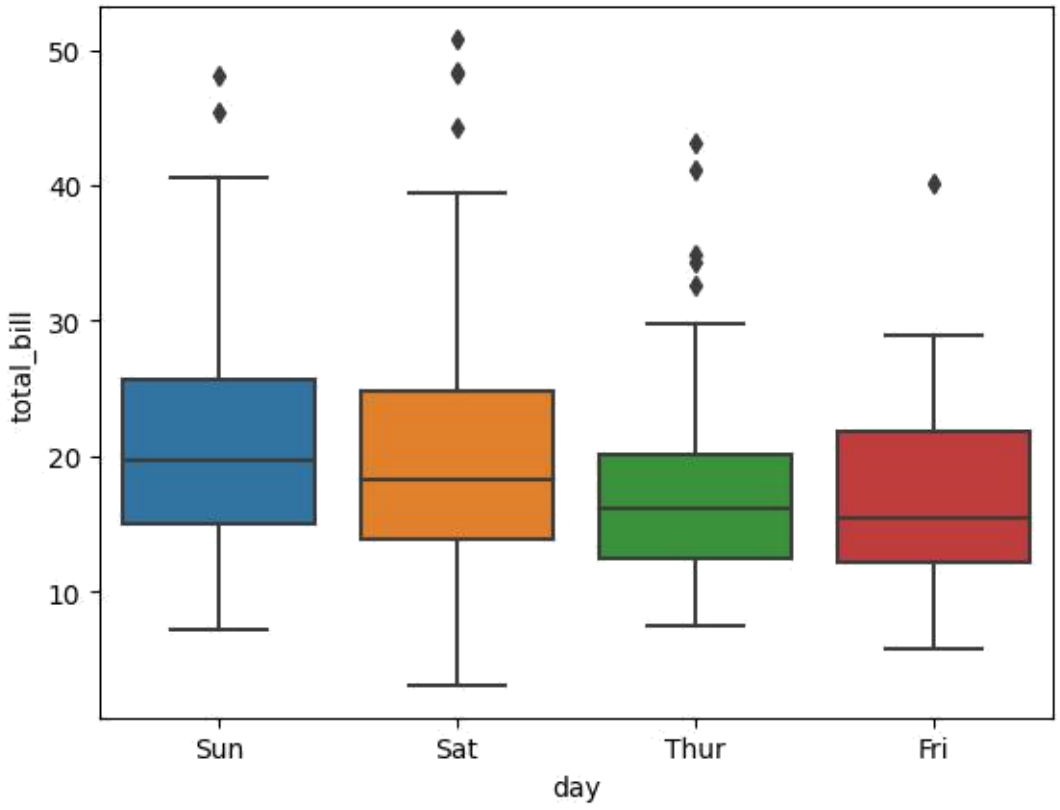
[12]:
```
# box plot is a stranderdized way of displaying the distribution of
 data. bases␣ on five number summary.
# it can also tell your outliers and what there values are .
# box plots can also tell you if ypur data is symmetrical ,
# how tightly your data is grouped and if and how yor data is skewed .
```

[13]:
```
sns.boxplot(data=data2,y="total_bill")
```

[13]: <Axes: ylabel='total_bill'>



[14]:
```
sns.boxplot(data=data2,x="day",y="total_bill")
plt.show()
```



[15]:
```
data1.head()
```

[15]:
```
   Sepal Length  Sepal Width Petal Length  Petal Width        iris  \
0           5.1          3.5           1.4          0.2  Iris-setosa
1           4.9          3.0           1.4          0.2  Iris-setosa
2           4.7          3.2           1.3          0.2  Iris-setosa
3           4.6          3.1           1.5          0.2  Iris-setosa
4           5.0          3.6           1.4          0.2  Iris-setosa

   species no
0          1
1          1
2          1
3          1
4          1
```

[16]:
```
data2.head()
```

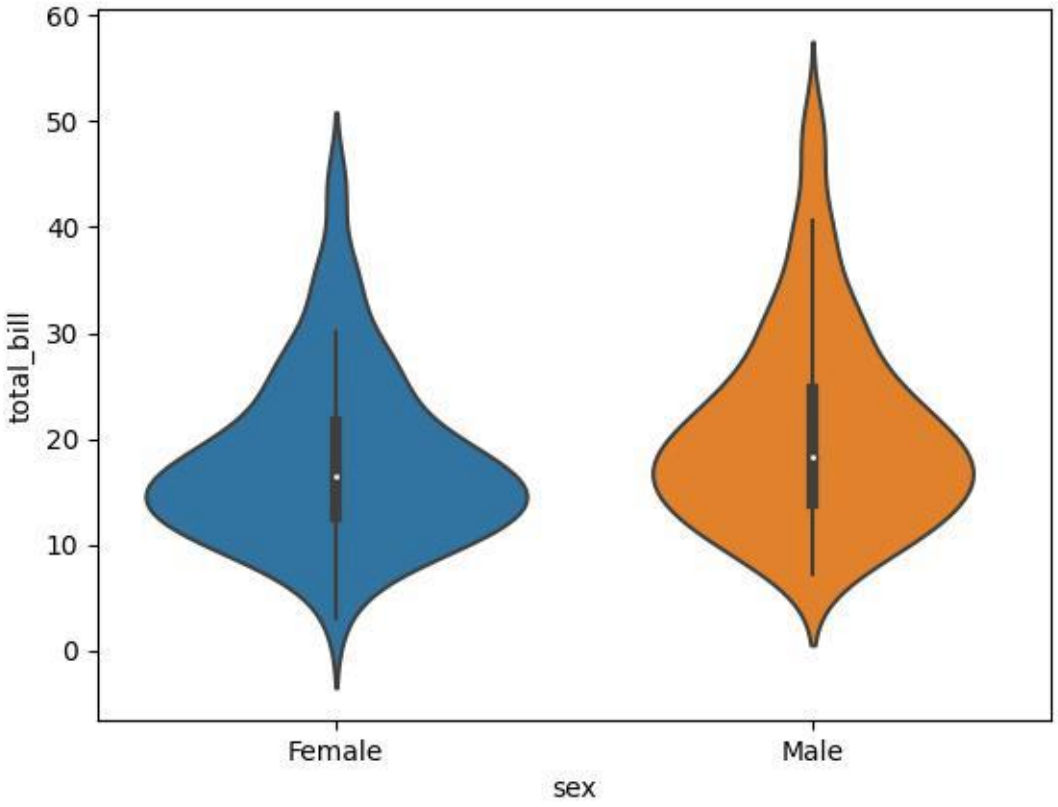
[16]:
```
   total_bill   tip    sex smoker day    time size
0       16.99  1.01 Female     No Sun  Dinner    2
1       10.34  1.66   Male     No Sun  Dinner    3
```

```
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
```

[17]:
```
# viloin plot
```

[18]:
```
sns.violinplot(data=data2,x="sex",y="total_bill")
```

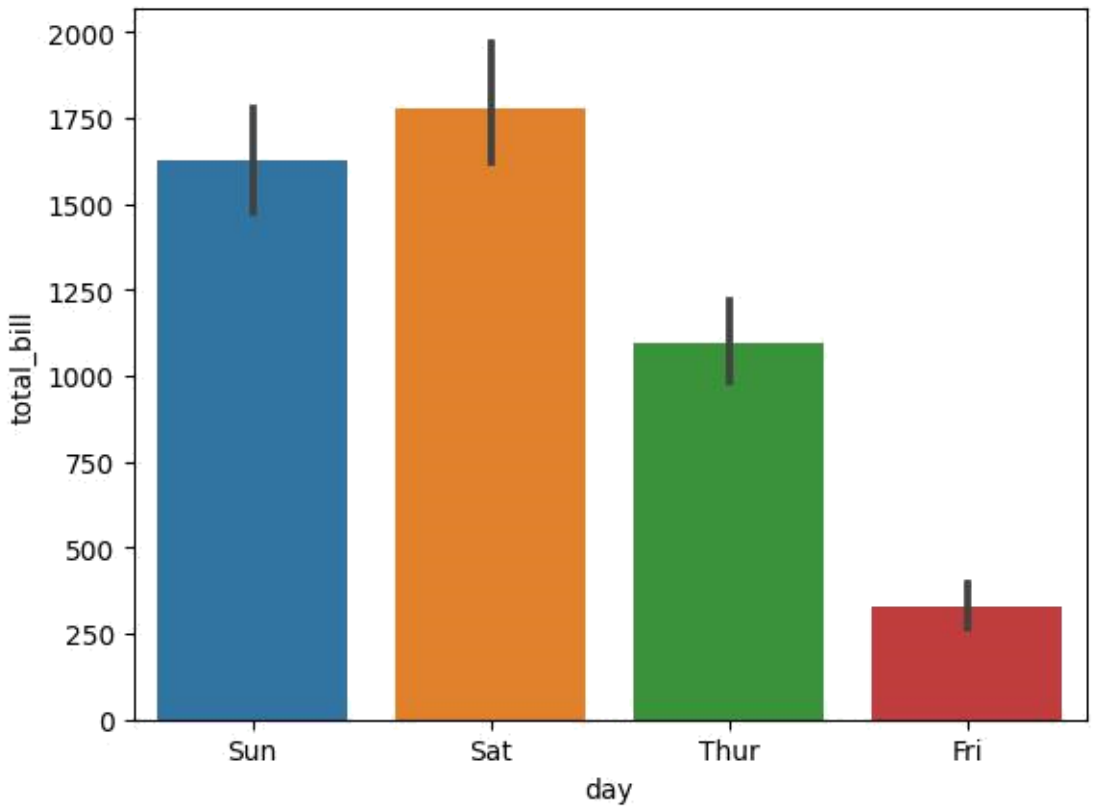
[18]: <Axes: xlabel='sex', ylabel='total_bill'>



[19]:
```
# barplot
```

[27]:
```
sns.barplot(data=data2,x="day",y="total_bill",estimator="sum")
```

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

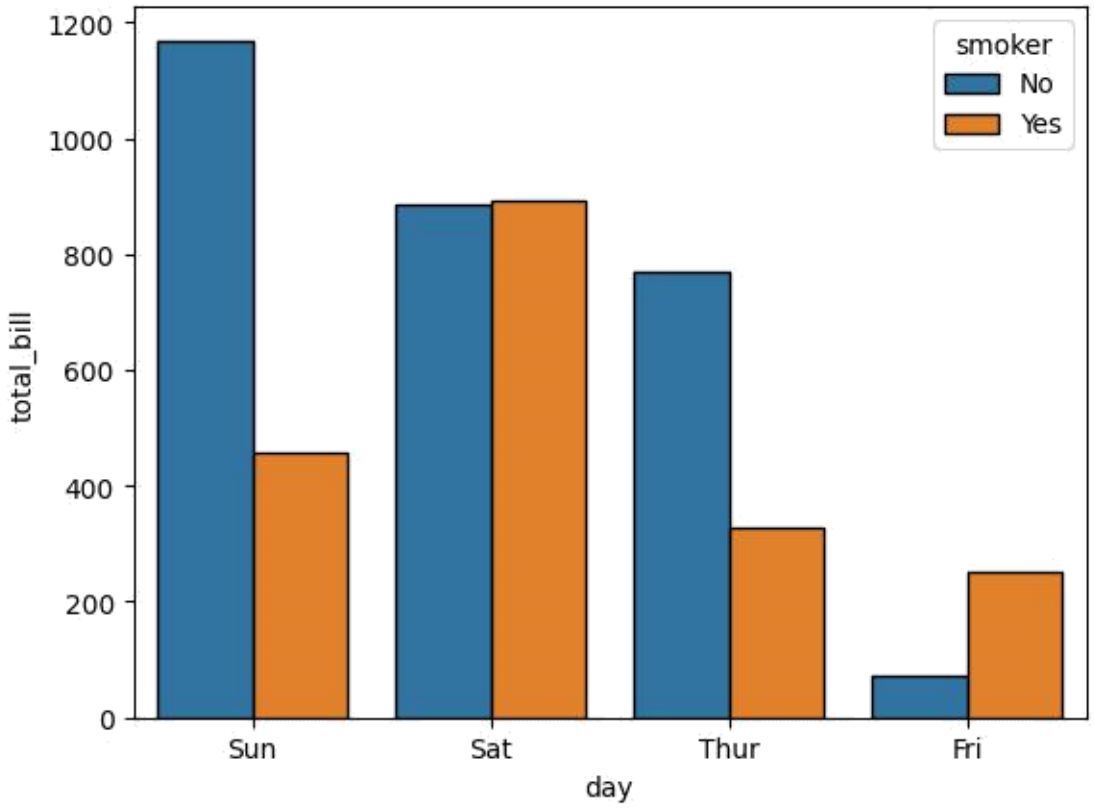


[26]:
```
data2[data2["day"]=="Sun"]["total_bill"].mean()
```

[26]: 21.41

[30]:
```
sns.
barplot(data=data2,x="day",y="total_bill",estimator="sum",hue="smoker",errorbar=None,edgeco
```

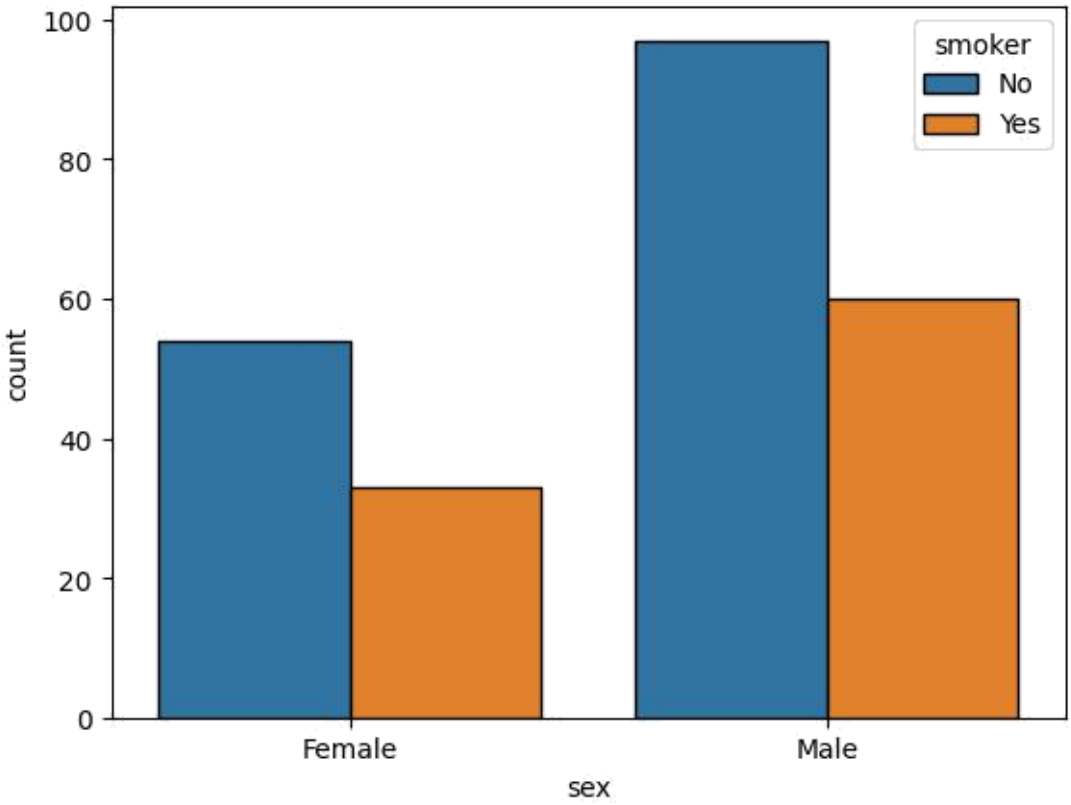
[30]: <Axes: xlabel='day', ylabel='total_bill'>



[31]:
```
# countplot
```

[39]:
```
sns.countplot(data=data2,x="sex",edgecolor="black",hue="smoker")
```

[39]: <Axes: xlabel='sex', ylabel='count'>



[40]:
```
data2
```

[40]:

	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

[244 rows x 7 columns]

[45]:
```
temp = data2[data2["day"]=="Sun"]
```

[53]:
```
temp["total_bill"].values
```

[53]:
```
array([16.99, 10.34, 21.01, 23.68, 24.59, 25.29,  8.77, 26.88,
                                                    15.04,
       14.78, 10.27, 35.26, 15.42, 18.43, 14.83, 21.58, 10.33, 16.29,
       16.97, 17.46, 13.94,                9.68, 30.4 , 18.29, 22.23, 32.4 ,
                                                    28.55,
       18.04, 12.54, 10.29, 34.81,  9.94, 25.56, 19.49, 38.07, 23.95,
       25.71, 17.31, 29.93, 14.07, 13.13, 17.26, 24.55, 19.77, 29.85,
       48.17, 25.                                 , 13.39, 16.49, 21.5 ,
                                             12.66, 16.21, 13.81, 17.51,
       24.52, 20.76, 31.71,                7.25, 31.85, 16.82, 32.9 , 17.89,
                                                    14.48,
        9.6 , 34.63, 34.65, 23.33, 45.35, 23.17, 40.55, 20.69, 20.9 ,
              30.46, 18.15, 23.1 , 15.69])
```

[55]:
```
data2
```

[55]:

	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

[244 rows x 7 columns]

[58]:
```
data2[(data2["sex"]=="Male") & (data2["smoker"]=="No")]
```

[58]:

	total_bill	tip	sex	smoker	day	time	size
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
5	25.29	4.71	Male	No	Sun	Dinner	4
6	8.77	2.00	Male	No	Sun	Dinner	2
..	…	…	…	…	…	…	…
232	11.61	3.39	Male	No	Sat	Dinner	2
233	10.77	1.47	Male	No	Sat	Dinner	2
235	10.07	1.25	Male	No	Sat	Dinner	2
239	29.03	5.92	Male	No	Sat	Dinner	3
242	17.82	1.75	Male	No	Sat	Dinner	2

[97 rows x 7 columns]

[59]:
```
data2["sex"]
```

```
[59]: 0      Female
      1        Male
      2        Male
      3        Male
      4      Female
              ...
      239      Male
      240    Female
      241      Male
      242      Male
      243    Female
      Name: sex, Length: 244, dtype: object
```

```
[60]: data2.sex
```

```
[60]: 0      Female
      1        Male
      2        Male
      3        Male
      4      Female
              ...
      239      Male
      240    Female
      241      Male
      242      Male
      243    Female
      Name: sex, Length: 244, dtype: object
```

```
[61]: data2.day
```

```
[61]: 0       Sun
      1       Sun
      2       Sun
      3       Sun
      4       Sun
             ...
      239     Sat
      240     Sat
      241     Sat
      242     Sat
      243    Thur
      Name: day, Length: 244, dtype: object
```

```
[70]: data2[(data2.sex=="Female") & (data2.smoker=="Yes") &
        (data2.time=="Dinner") & (data2.total_bill>=20)]
```

[70]:

	total_bill	tip	sex	smoker	day	time	size
72	26.86	3.14	Female	Yes	Sat	Dinner	2
73	25.28	5.00	Female	Yes	Sat	Dinner	2
102	44.30	2.50	Female	Yes	Sat	Dinner	3
103	22.42	3.48	Female	Yes	Sat	Dinner	2
186	20.90	3.50	Female	Yes	Sun	Dinner	3
214	28.17	6.50	Female	Yes	Sat	Dinner	3
219	30.14	3.09	Female	Yes	Sat	Dinner	4
229	22.12	2.88	Female	Yes	Sat	Dinner	2
240	27.18	2.00	Female	Yes	Sat	Dinner	2

[65]:
```
print(int(True))
```

```
1
```

[66]:
```
print(True + 1)
```

```
2
```

[67]:
```
print(int(True) + 1)
```

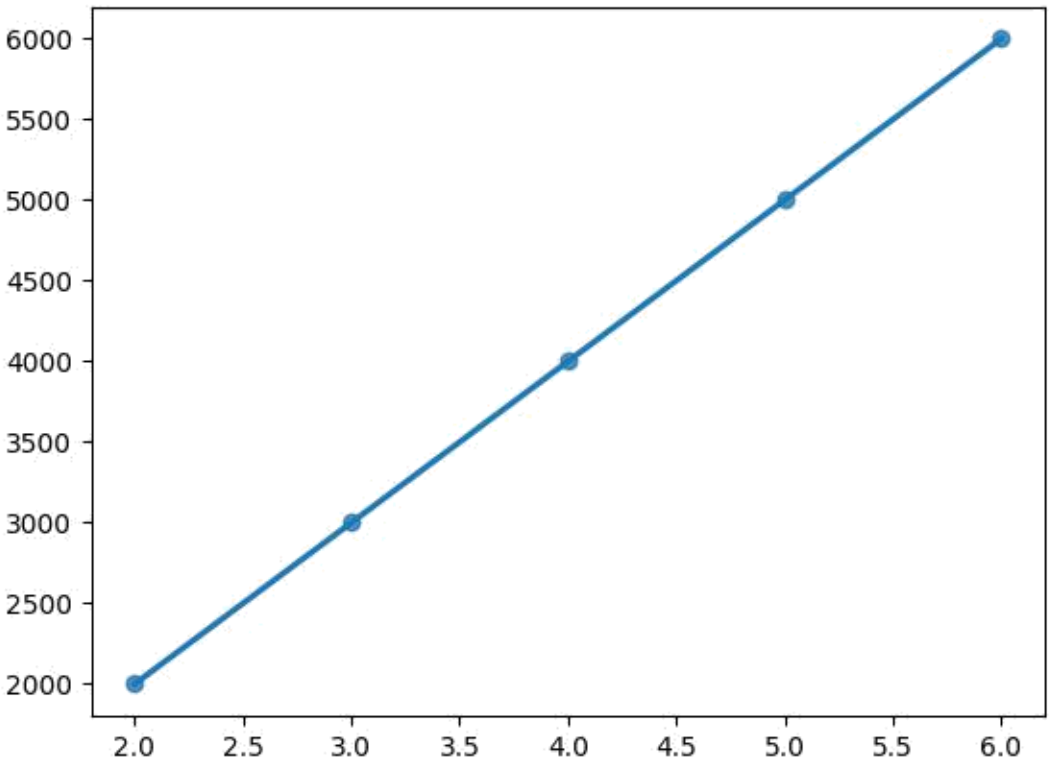
```
2
```

[71]:
```
# Regression plot
```

[76]:
```
exp = [2,3,4,5,6]
salary = [2000,3000,4000,5000,6000]

sns.regplot(x=exp,y=salary)
```

[76]: <Axes: >



[74]:
```
data2
```

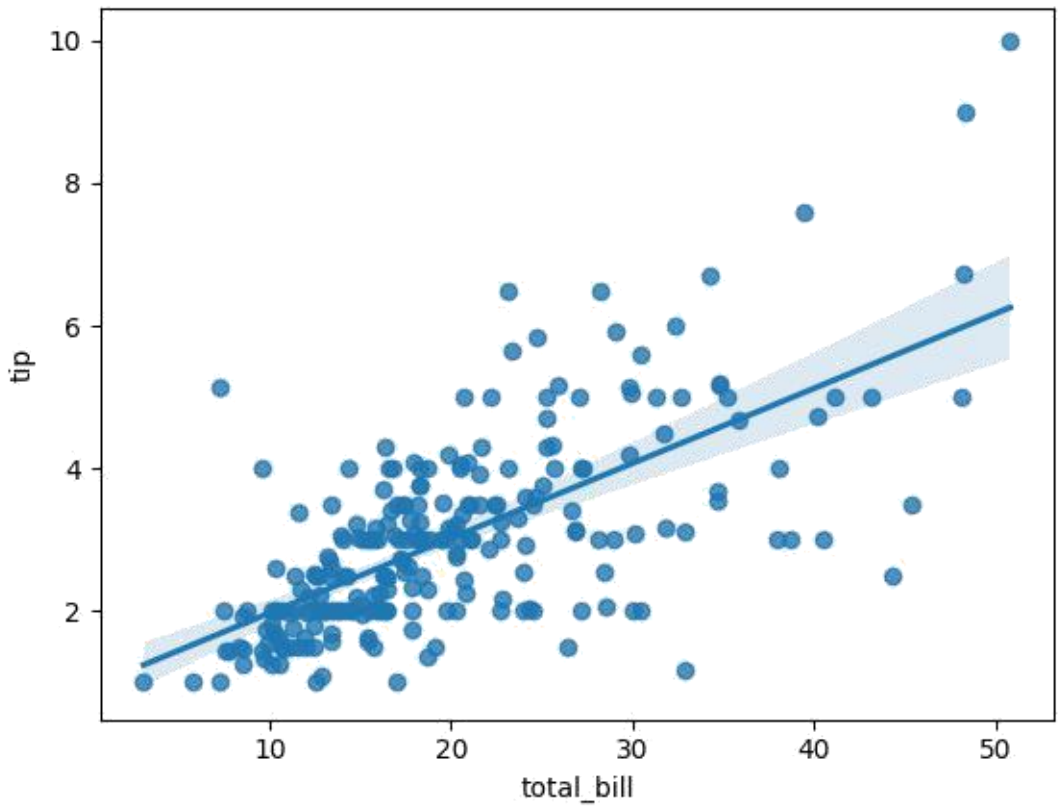
[74]:

	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

[244 rows x 7 columns]

[75]:
```
sns.regplot(data=data2,x="total_bill",y="tip")
```

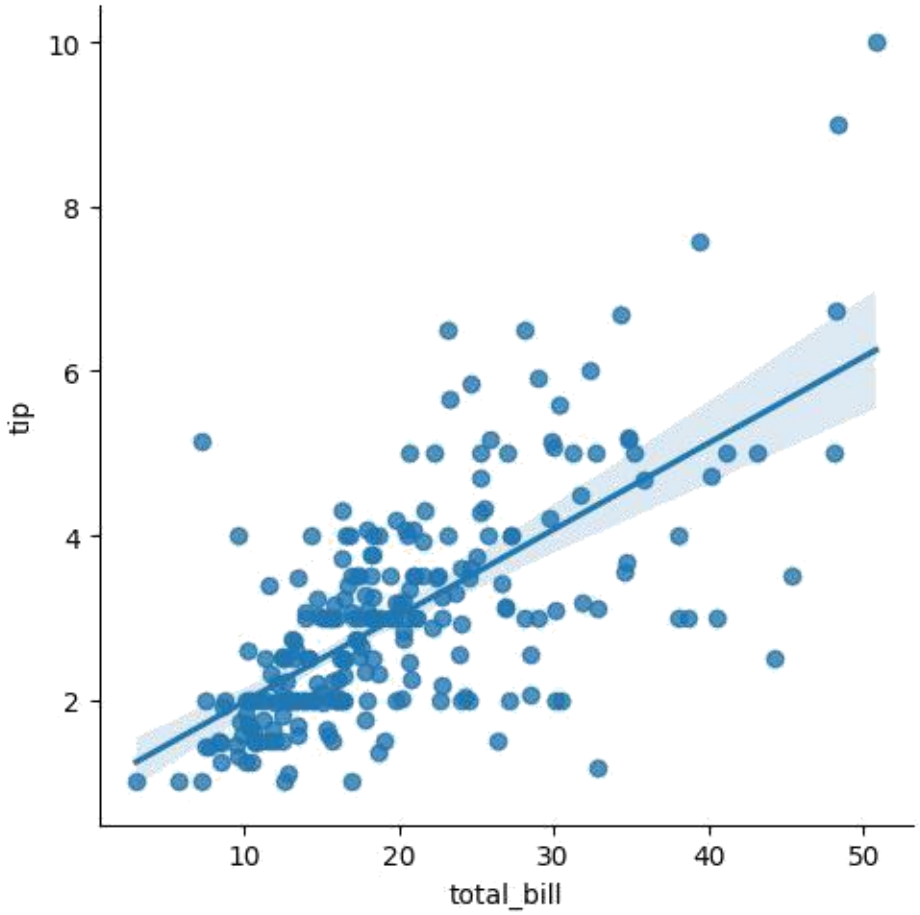
[75]: <Axes: xlabel='total_bill', ylabel='tip'>



[78]:
```
sns.lmplot(data=data2,x="total_bill",y="tip")
```

```
C:\Users\Admin\AppData\Local\Programs\Python\Python311\Lib\site-
packages\seaborn\axisgrid.py:118: UserWarning: The figure layout has
changed to tight
  self._figure.tight_layout(*args, **kwargs)
```

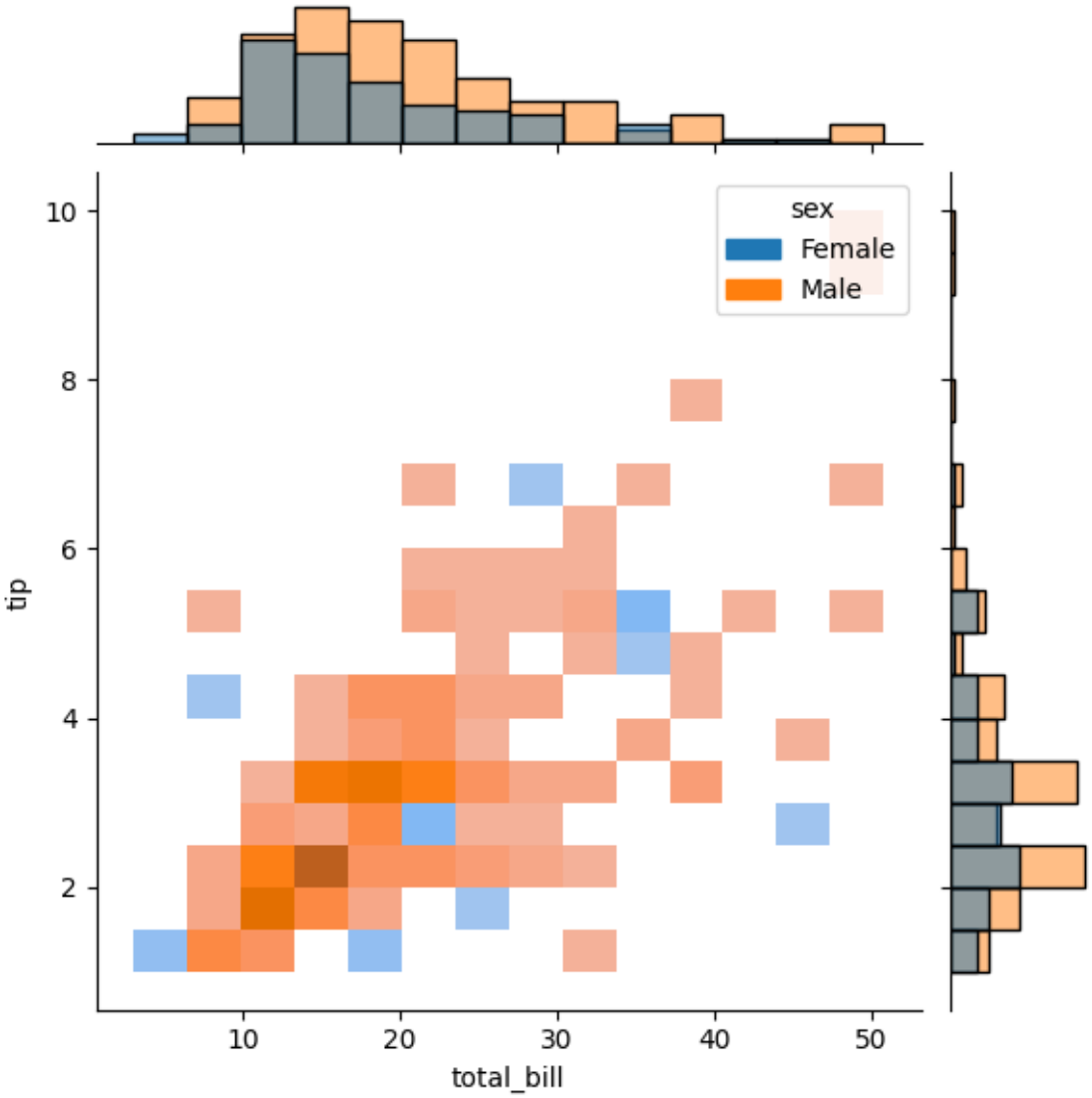
[78]: <seaborn.axisgrid.FacetGrid at 0x1e9bcdb61d0>



[80]:
```
# 6.Jointplot -> is a stylish way to plot graphs .
```

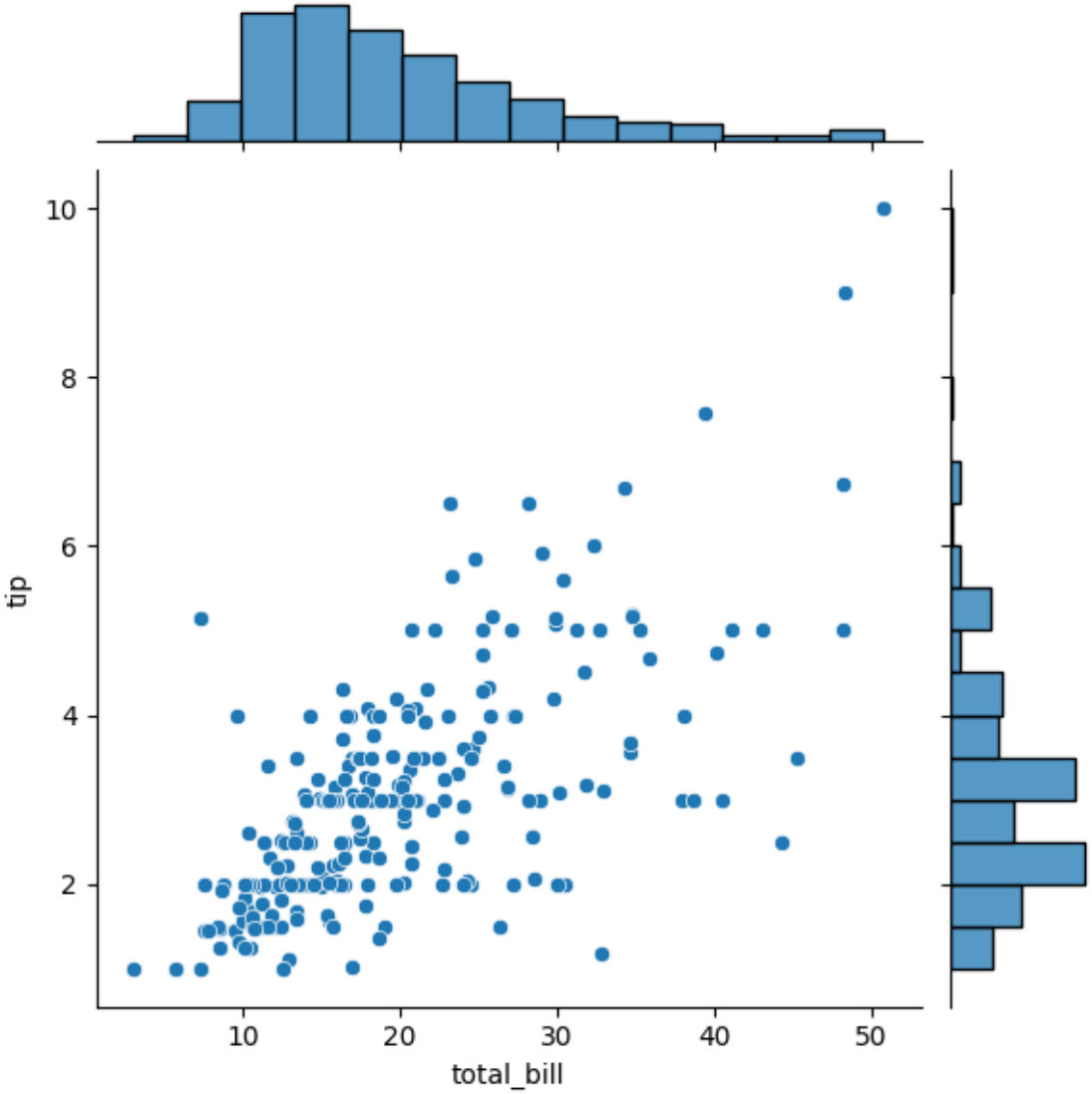
[83]:
```
sns.jointplot(data=data2,x="total_bill",y="tip",hue="sex",kind="hist")
```

[83]: <seaborn.axisgrid.JointGrid at 0x1e9beaa81d0>



[90]:
```
g = sns.JointGrid(data=data2,x="total_bill",y="tip")
g.plot(sns.scatterplot,sns.histplot)
```

[90]: <seaborn.axisgrid.JointGrid at 0x1e9c0568950>



[91]:
```
data2
```

[91]:

	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

[244 rows x 7 columns]
```

[99]:
```
gap = pd.read_csv("datasets/gapminder.csv")
```

[100]:
```
gap
```

[100]:
```
       country continent  year  life_exp  hdi_index  co2_consump     gdp  \
0     Afghanistan      Asia  1998      53.3      0.344       0.0522     NaN
1     Afghanistan      Asia  1999      54.7      0.348       0.0402     NaN
2     Afghanistan      Asia  2000      54.7      0.350       0.0370     NaN
3     Afghanistan      Asia  2001      54.8      0.353       0.0376     NaN
4     Afghanistan      Asia  2002      55.5      0.384       0.0471   333.0
…             …       …     …         …          …            …      …
3670    Zimbabwe    Africa  2014      58.0      0.547       0.8810  1440.0
3671    Zimbabwe    Africa  2015      58.6      0.553       0.8810  1450.0
3672    Zimbabwe    Africa  2016      59.2      0.558       0.7710  1430.0
3673    Zimbabwe    Africa  2017      59.9      0.563       0.8450  1480.0
3674    Zimbabwe    Africa  2018      60.6      0.569       0.8500  1510.0

      services
0         24.4
1         24.6
2         24.7
3         24.7
4         25.6
……
367025.4
367125.7
367226.1
367326.6
367427.2

[3675 rows x 8 columns]
```

[109]:
```
temp = gap.pivot_table(index="country",columns="year",values="life_exp").
  head(10)
```

[110]:
```
temp
```

[110]:
```
year         1998  1999  2000  2001  2002  2003  2004  2005  2006  2007 …  \
country                                                                    …
Afghanistan  53.3  54.7  54.7  54.8  55.5  56.5  57.1  57.6  58.0  58.5 …
Albania      74.8  75.1  75.4  76.0  75.9  75.6  75.8  76.2  76.9  77.5 …
Algeria      70.2  70.7  71.0  71.3  71.8  72.0  72.6  72.9  73.3  73.6 …
Angola       50.6  51.9  52.8  53.4  54.5  55.1  55.5  56.4  57.0  58.0 …
```

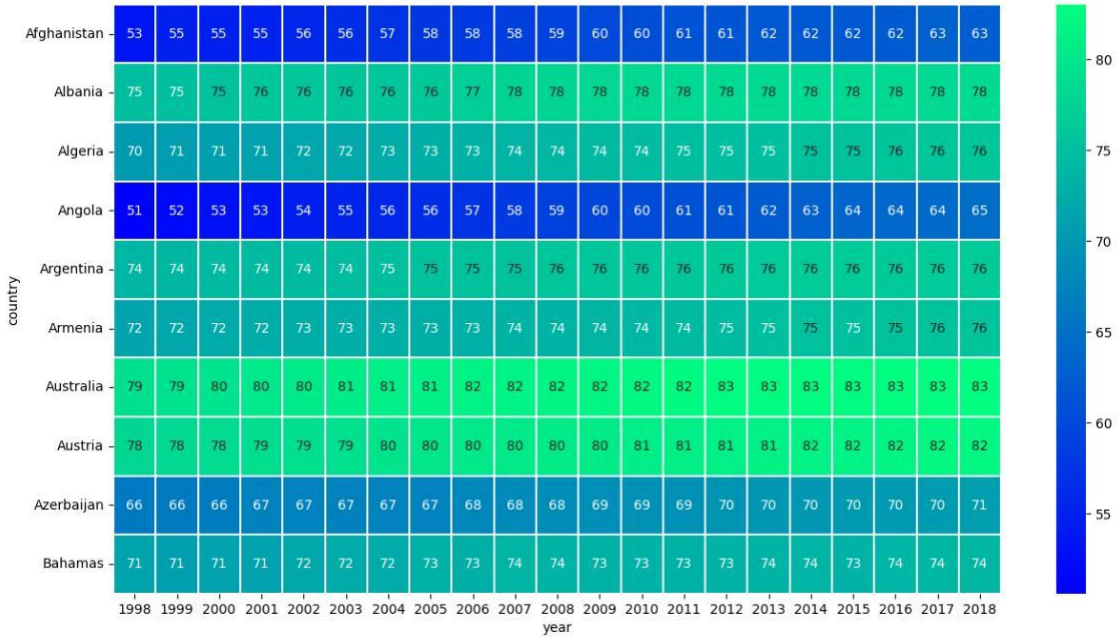
```
Argentina   73.7 73.8 74.2 74.3 74.3 74.4 74.9 75.3 75.4 75.3 …
Armenia     71.6 71.9 72.4 72.5 72.7 72.8 73.0 73.0 73.1 73.5 …
Australia   79.1 79.4 79.7 80.1 80.3 80.6 80.9 81.2 81.5 81.5 …
Austria     77.9 78.2 78.5 78.9 79.0 79.1 79.5 79.8 80.1 80.3 …
Azerbaijan  66.0 66.2 66.5 67.1 67.2 67.1 67.2 67.3 67.7 68.2 …
Bahamas     71.4 70.8 71.3 71.3 71.9 72.4 72.4 73.2 73.0 73.6 …

year        2009 2010 2011 2012 2013 2014 2015 2016 2017 2018
country
Afghanistan 59.9 60.5 61.0 61.4 61.9 61.9 61.9 62.0 62.9 62.7
Albania     78.0 78.1 78.1 78.2 78.3 78.2 78.1 78.2 78.3 78.4
Algeria     74.2 74.5 74.7 74.9 75.1 75.3 75.4 75.7 75.9 76.0
Angola      59.5 60.2 60.8 61.4 62.1 63.0 63.5 63.9 64.2 64.6
Argentina   75.8 75.9 76.0 76.2 76.3 76.5 76.5 76.2 76.3 76.5
Armenia     73.6 73.9 74.2 74.6 75.1 75.2 75.1 75.3 75.5 75.6
Australia   81.9 82.1 82.3 82.6 82.7 82.7 82.7 83.0 83.0 82.9
Austria     80.5 80.8 81.0 81.2 81.3 81.5 81.6 81.8 82.0 82.1
Azerbaijan  68.8 69.0 69.2 69.5 69.7 69.9 70.2 70.3 70.4 70.8
Bahamas     73.2 73.2 73.2 73.3 73.5 73.6 73.3 73.7 73.8 73.8

[10 rows x 21 columns]
```

[115]:
```
plt.figure(figsize=(15,8))
sns.heatmap(data=temp,annot=True,linewidth=0.1,cmap="winter")
```

[115]: <Axes: xlabel='year', ylabel='country'>



[ ]: