In [1]:

```
import pandas as pd
import matplotlib.pyplot as plt
import seaborn as sns
```

In [2]:

```
df=sns.load_dataset("tips")
df
```

Out[2]:

	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 × 7 columns

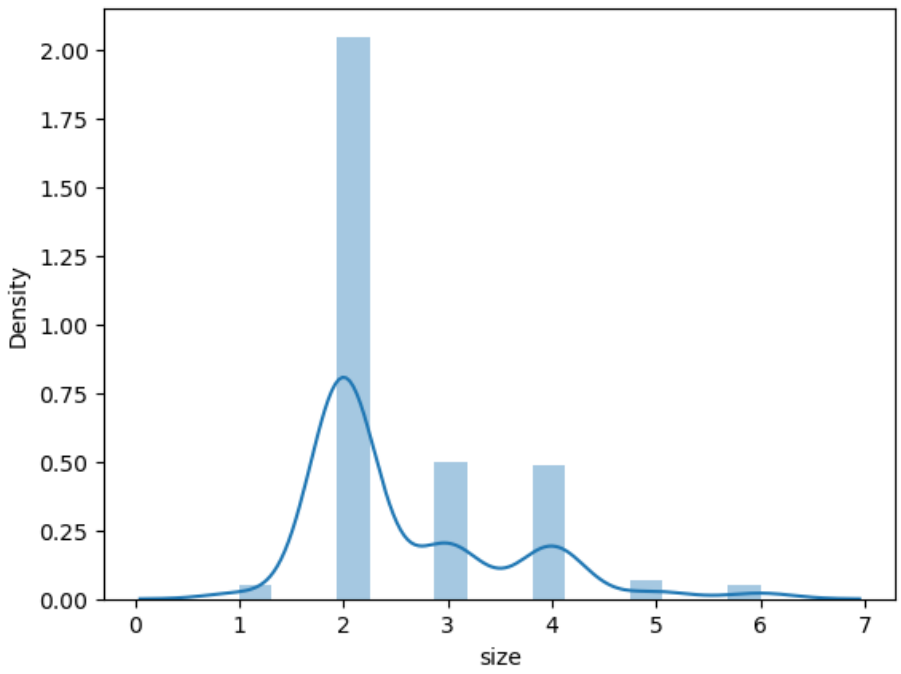
In [3]:

```
sns.distplot(df['size'])
```

```
C:\ProgramData\Anaconda3\lib\site-packages\seaborn\distributions.py:2619: FutureWarning: `distplo
t` is a deprecated function and will be removed in a future version. Please adapt your code to us
e either `displot` (a figure-level function with similar flexibility) or `histplot` (an axes-leve
l function for histograms).
  warnings.warn(msg, FutureWarning)
```

Out[3]:

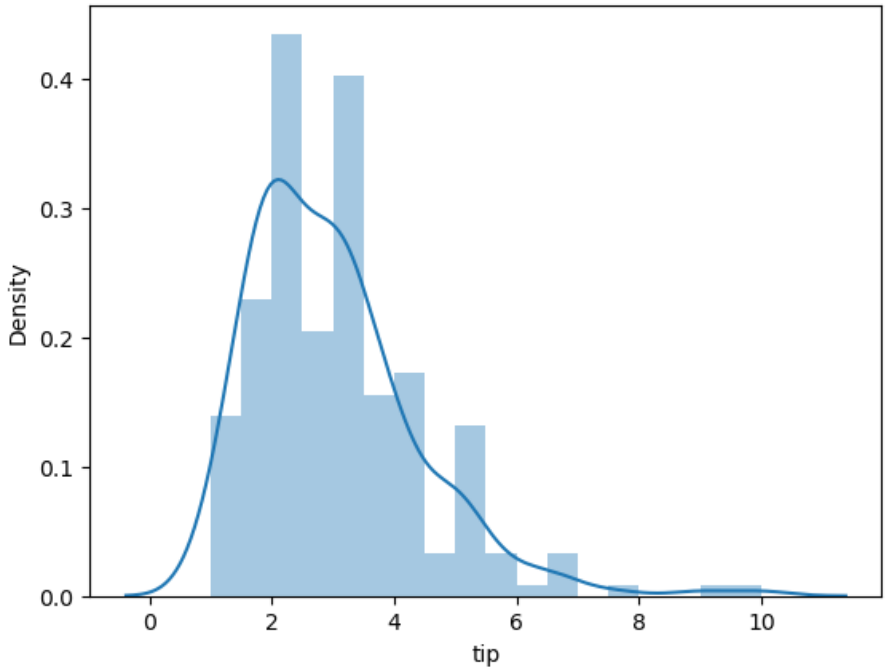
```
<AxesSubplot:xlabel='size', ylabel='Density'>
```



In [4]:

```
sns.distplot(df['tip'])
plt.show()
```

```
C:\ProgramData\Anaconda3\lib\site-packages\seaborn\distributions.py:2619: FutureWarning: `distplo
t` is a deprecated function and will be removed in a future version. Please adapt your code to us
e either `displot` (a figure-level function with similar flexibility) or `histplot` (an axes-leve
l function for histograms).
  warnings.warn(msg, FutureWarning)
```



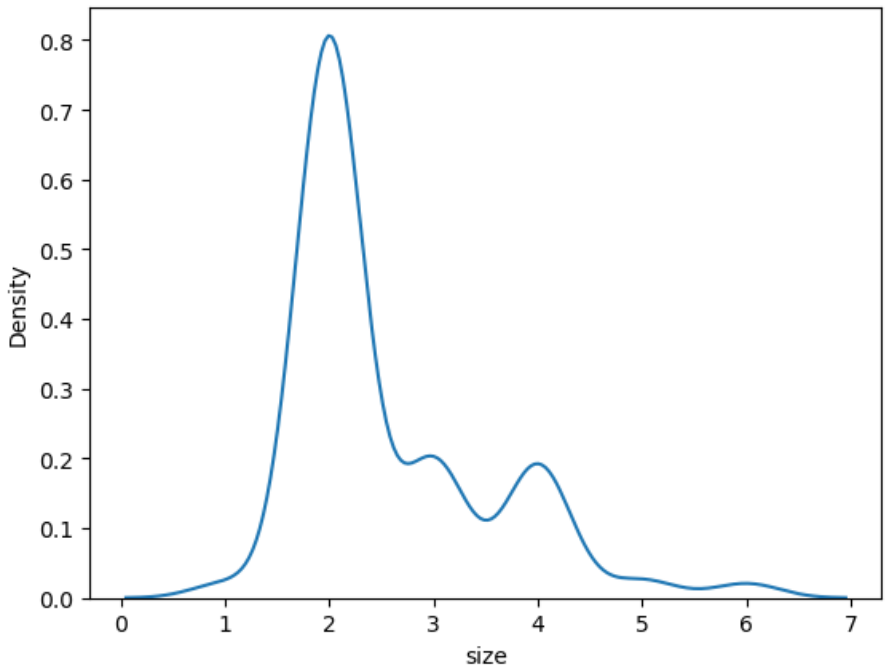
In [5]:

```
sns.distplot(df['size'],hist=False)
```

```
C:\ProgramData\Anaconda3\lib\site-packages\seaborn\distributions.py:2619: FutureWarning: `distplo
t` is a deprecated function and will be removed in a future version. Please adapt your code to us
e either `displot` (a figure-level function with similar flexibility) or `kdeplot` (an axes-level
function for kernel density plots).
  warnings.warn(msg, FutureWarning)
```

Out[5]:

```
<AxesSubplot:xlabel='size', ylabel='Density'>
```



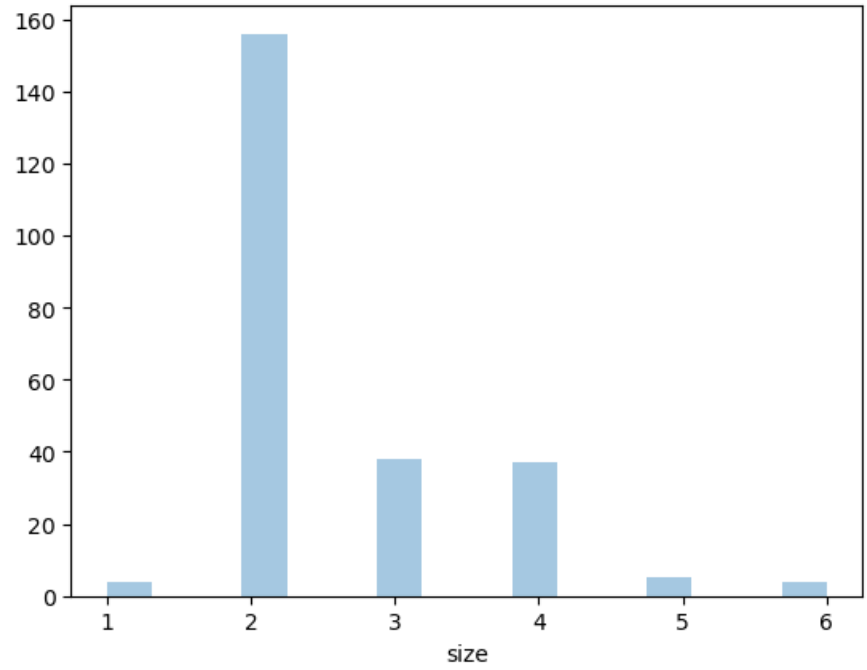
In [6]:

```
sns.distplot(df['size'],kde=False)
```

```
C:\ProgramData\Anaconda3\lib\site-packages\seaborn\distributions.py:2619: FutureWarning: `distplo
t` is a deprecated function and will be removed in a future version. Please adapt your code to us
e either `displot` (a figure-level function with similar flexibility) or `histplot` (an axes-leve
l function for histograms).
  warnings.warn(msg, FutureWarning)
```

Out[6]:

```
<AxesSubplot:xlabel='size'>
```

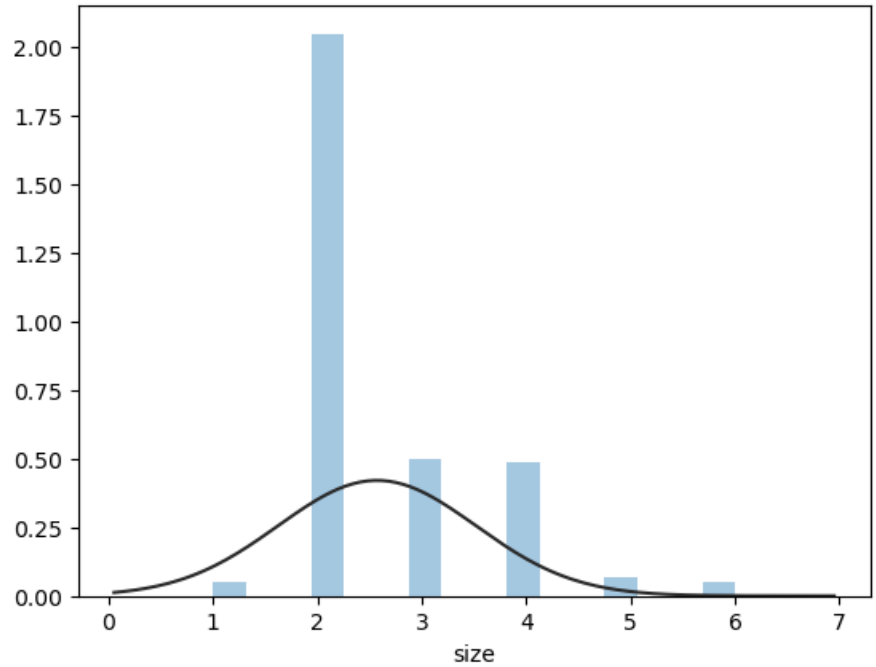


In [7]:

```
from scipy.stats import norm
sns.distplot(df['size'],fit=norm,kde=False)
```

Out[7]:

```
<AxesSubplot:xlabel='size'>
```



In [8]:

```
df1=pd.read_csv(r"C:\Users\user\Desktop\Product1.csv")
df1
```

Out[8]:

	Order_ID	Order_Date	Customer_Name	Salesperson	Region	Payment_Type	Category	Unit_Price	Quantity	Revenue
0	1001	2014-01-27	CompanyAA	Mariya Sergienko	West	Check	Beverages	14.0	49	686.0
1	1002	2014-01-27	CompanyAA	Mariya Sergienko	West	Check	Dried Fruit & Nuts	3.5	47	164.5
2	1003	2014-01-04	Company D	Andrew Cencini	East	Credit Card	Dried Fruit & Nuts	30.0	69	2070.0
3	1004	2014-01-04	Company D	Andrew Cencini	East	Credit Card	Dried Fruit & Nuts	53.0	89	4717.0
4	1005	2014-01-04	Company D	Andrew Cencini	East	Credit Card	Dried Fruit & Nuts	3.5	11	38.5
...	...	...	...	...	...	...	...	...	...	...
384	1067	2014-03-08	Company H	Nancy Freehafer	North	Credit Card	Dairy Products	34.8	63	2192.4
385	1070	2014-03-03	Company C	Mariya Sergienko	West	Cash	Condiments	10.0	48	480.0
386	1071	2014-03-03	Company C	Mariya Sergienko	West	Cash	Sauces	40.0	71	2840.0
387	1075	2014-03-10	Company J	Laura Giussani	East	Credit Card	Dried Fruit & Nuts	10.0	55	550.0
388	1077	2014-03-10	Company J	Laura Giussani	East	Cash	Dried Fruit & Nuts	3.5	21	73.5

389 rows × 12 columns

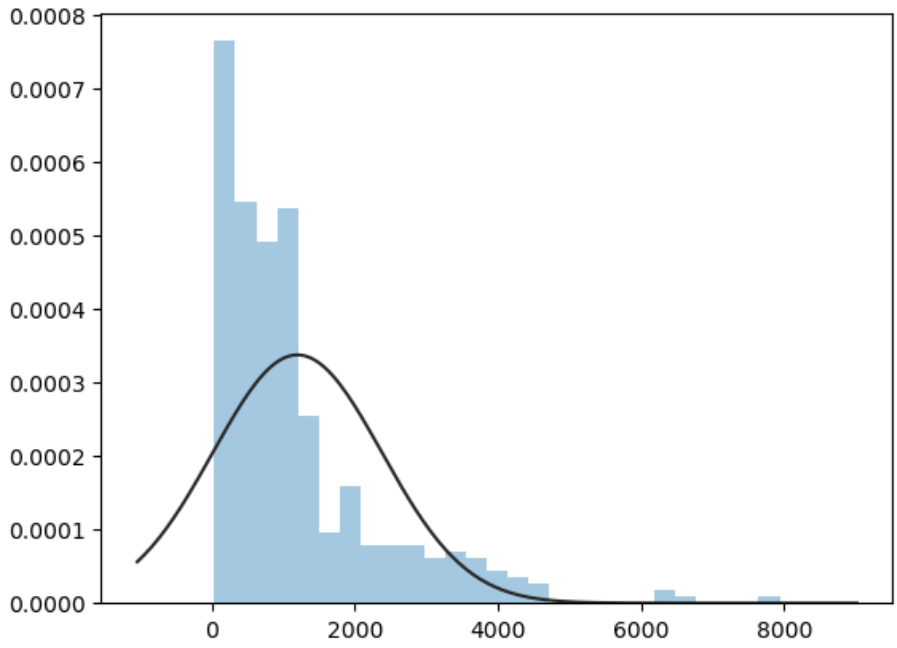
In [9]:

```
sns.distplot(df1[['Revenue']],fit=norm,kde=False)
```

```
C:\ProgramData\Anaconda3\lib\site-packages\seaborn\distributions.py:2619: FutureWarning: `distplo
t` is a deprecated function and will be removed in a future version. Please adapt your code to us
e either `displot` (a figure-level function with similar flexibility) or `histplot` (an axes-leve
l function for histograms).
  warnings.warn(msg, FutureWarning)
```

Out[9]:

```
<AxesSubplot:>
```



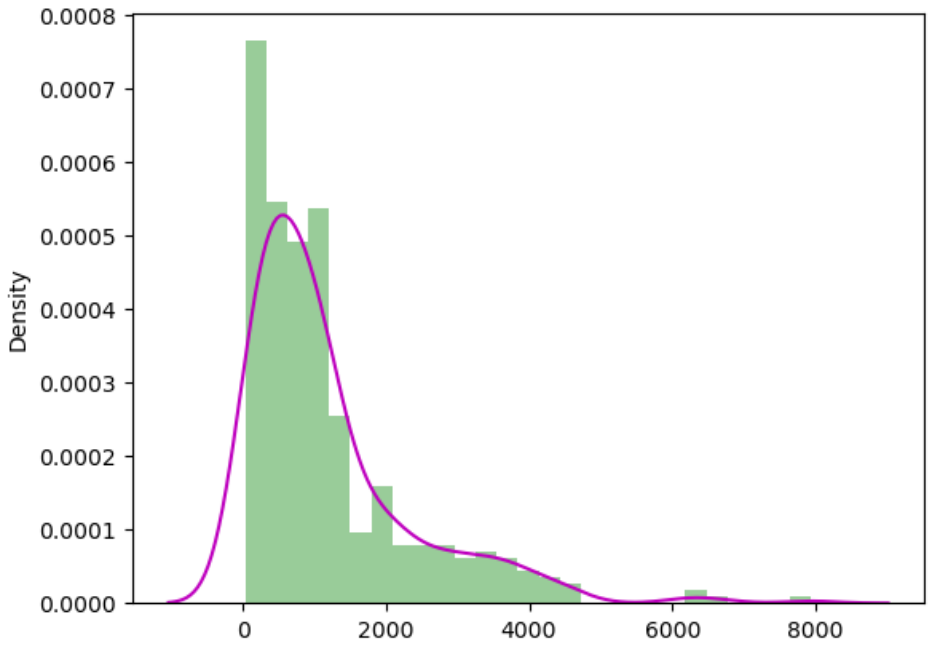
In [10]:

```
kde1={"color": "m"}
hist1={"color": "g"}
sns.distplot(df1[['Revenue']],kde_kws=kde1,hist_kws=hist1)
```

```
C:\ProgramData\Anaconda3\lib\site-packages\seaborn\distributions.py:2619: FutureWarning: `distplo
t` is a deprecated function and will be removed in a future version. Please adapt your code to us
e either `displot` (a figure-level function with similar flexibility) or `histplot` (an axes-leve
l function for histograms).
  warnings.warn(msg, FutureWarning)
```

Out[10]:

```
<AxesSubplot:ylabel='Density'>
```



Heatmap

In [11]:

```
import numpy as np
```

In [12]:

```
arr=np.linspace(1,5,12).reshape(3,4)
arr.shape
```

Out[12]:

```
(3, 4)
```

In [13]:

```
arr
```

Out[13]:

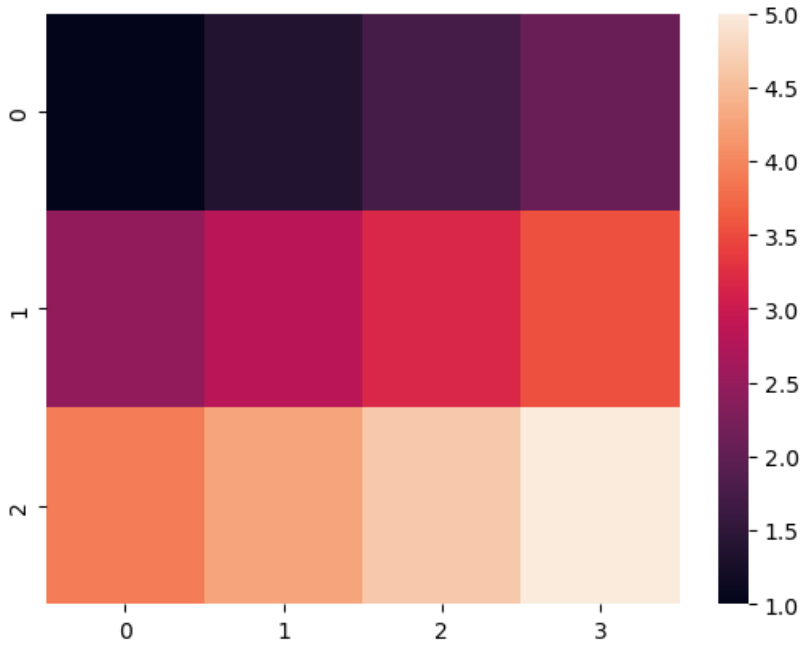
```
array([[1.        , 1.36363636, 1.72727273, 2.09090909],
       [2.45454545, 2.81818182, 3.18181818, 3.54545455],
       [3.90909091, 4.27272727, 4.63636364, 5.        ]])
```

In [14]:

```
sns.heatmap(arr)
```

Out[14]:

```
<AxesSubplot:>
```



In [17]:

```
globalWar=pd.read_csv(r"C:\Users\user\Desktop\Datasets\globalWarming.csv")
df3=globalWar.drop(columns=['Country Code','Indicator Name','Indicator Code'],axis=1).set_index("Country Name"
```

In [18]:

```
df3
```

Out[18]:

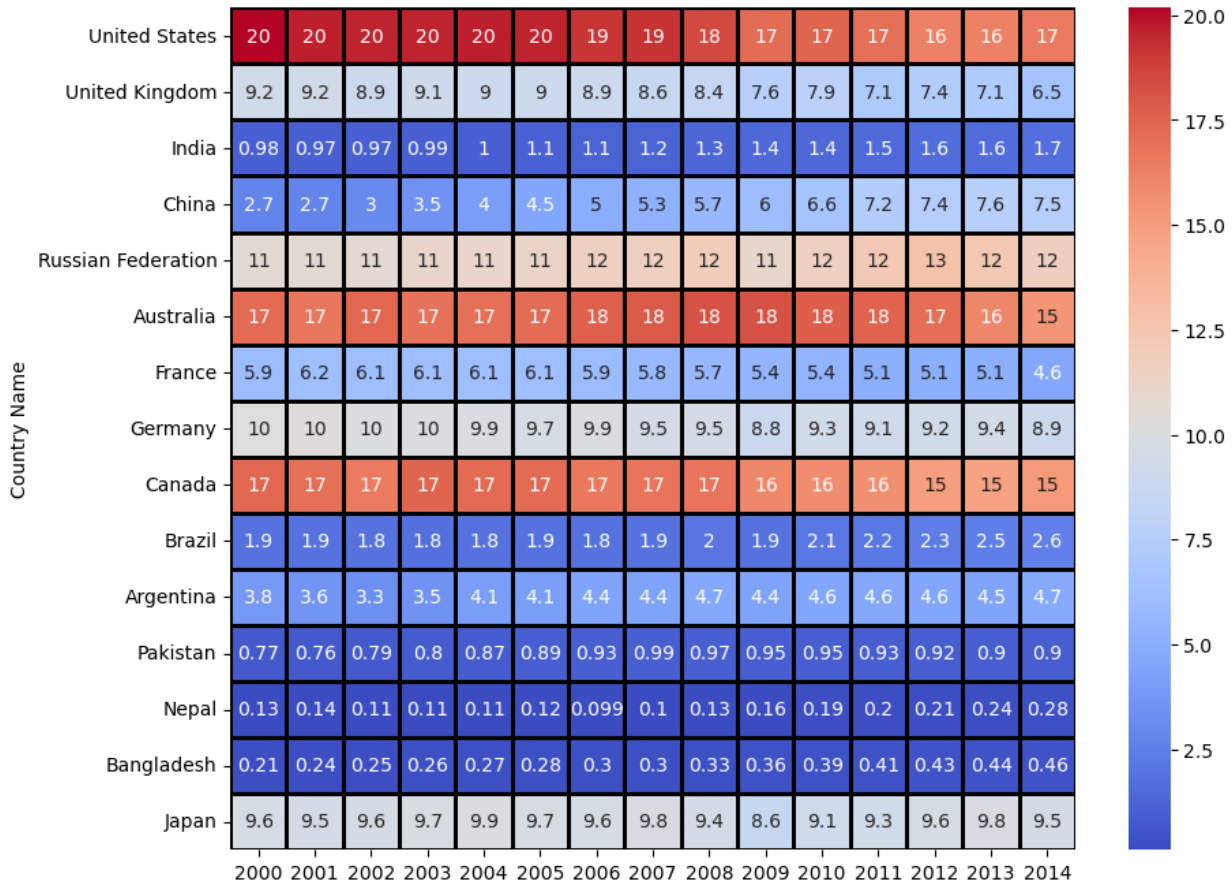
Country Name	2000	2001	2002	2003	2004	2005	2006	2007	2008	2009
United States	20.178751	19.636505	19.613404	19.564105	19.658371	19.591885	19.094067	19.217898	18.461764	17.157738
United Kingdom	9.199549	9.233175	8.904123	9.053278	8.989140	8.982939	8.898710	8.617164	8.424424	7.574622
India	0.979870	0.971698	0.967381	0.992392	1.025028	1.068563	1.121982	1.193210	1.310098	1.431844
China	2.696862	2.742121	3.007083	3.524074	4.037991	4.523178	4.980314	5.334910	5.701915	6.010102
Russian Federation	10.627121	10.669603	10.715901	11.090647	11.120627	11.253529	11.669122	11.672457	12.014507	11.023856
Australia	17.200610	16.733367	17.370452	16.901959	17.026515	17.169711	17.651398	17.865260	18.160876	18.200182
France	5.946665	6.153061	6.068664	6.115998	6.120079	6.099599	5.906266	5.766385	5.690501	5.438357
Germany	10.095640	10.366287	10.058673	9.969355	9.898682	9.666372	9.911476	9.488040	9.506321	8.818596
Canada	17.367115	16.985030	16.559378	17.461199	17.258911	17.251083	16.696694	16.855883	16.875198	15.961560
Brazil	1.871118	1.898354	1.844380	1.762482	1.828672	1.858088	1.839394	1.901372	2.008670	1.883812
Argentina	3.835574	3.568600	3.291548	3.525584	4.069058	4.141237	4.434821	4.382669	4.682912	4.410890
Pakistan	0.768458	0.764702	0.788668	0.804959	0.872802	0.887768	0.929857	0.991030	0.972050	0.950832
Nepal	0.129282	0.135226	0.106877	0.113902	0.105477	0.120277	0.098812	0.099736	0.129224	0.162087
Bangladesh	0.211802	0.242020	0.246756	0.256602	0.266823	0.275247	0.299529	0.301631	0.332728	0.357159
Japan	9.622352	9.464309	9.573130	9.725282	9.909203	9.698883	9.632049	9.782964	9.449534	8.620816

In [27]:

```
plt.figure(figsize=(10,8))
sns.heatmap(df3,cmap="coolwarm",annot=True,linewidths=1,
    linecolor='k')
```

Out[27]:

```
<AxesSubplot:ylabel='Country Name'>
```



In [28]:

```
df
```

Out[28]:

	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

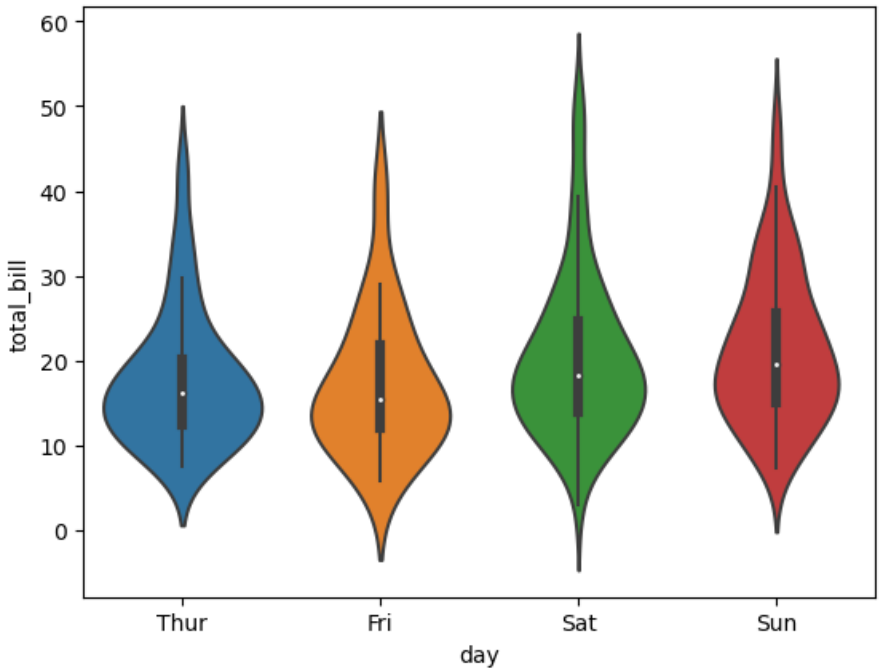
244 rows × 7 columns

In [29]:

```
sns.violinplot(x='day',y="total_bill",data=df)
```

Out[29]:

```
<AxesSubplot:xlabel='day', ylabel='total_bill'>
```

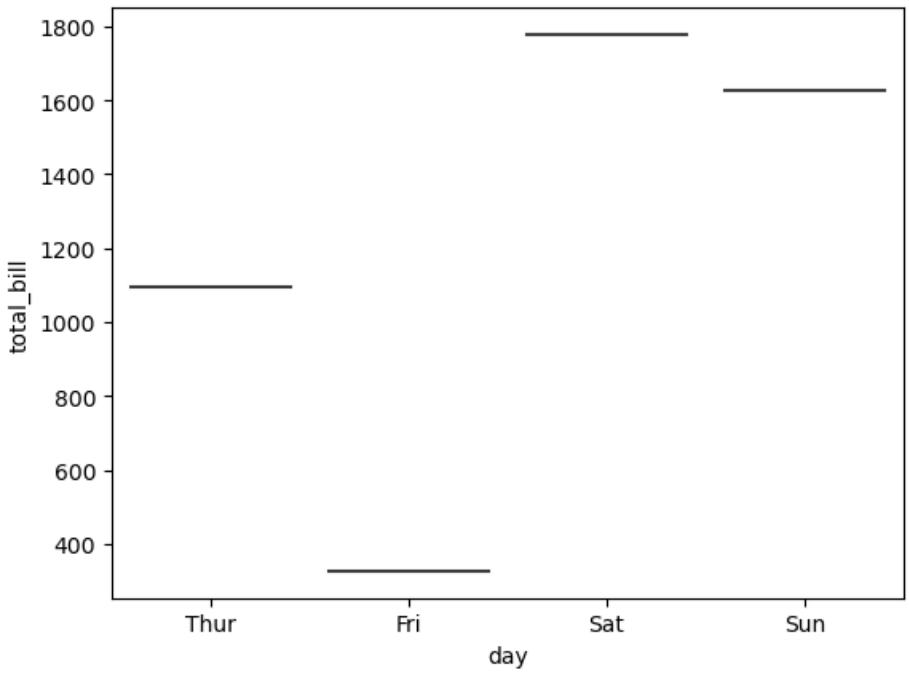


In [34]:

```
df4=df.groupby('day').sum()
df5=df4.sort_values(by="total_bill",ascending=False)
df5
sns.violinplot(x=df5.index,y='total_bill',data=df5)
```

Out[34]:

```
<AxesSubplot:xlabel='day', ylabel='total_bill'>
```

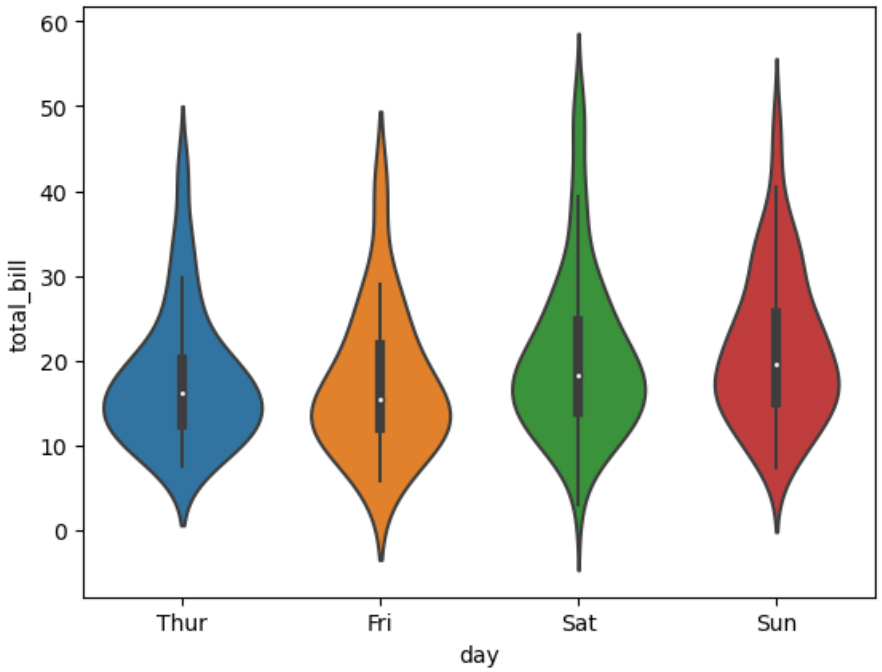


In [35]:

```
sns.violinplot(x='day',y="total_bill",data=df)
```

Out[35]:

```
<AxesSubplot:xlabel='day', ylabel='total_bill'>
```



In [ ]:

```

```