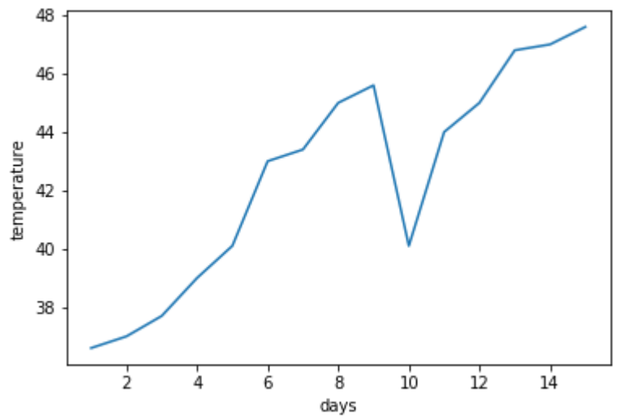
In [1]:
```
import numpy as np
import pandas as pd
import matplotlib.pyplot as plt
import seaborn as sns
```

In [2]:
```
# how to draw lineplot
days=[1,2,3,4,5,6,7,8,9,10,11,12,13,14,15]
temperature=[36.6,37,37.7,39,40.1,43,43.4,45,45.6,40.1,44,45,46.8,47,47.6]
data=pd.DataFrame({
    "days":days,
    "temperature":temperature
})
sns.lineplot(x="days",y="temperature",data=data)
plt.show()
```



In [3]:
```
# how to load csv file from online
df=sns.load_dataset("tips")
df
```

Out[3]:

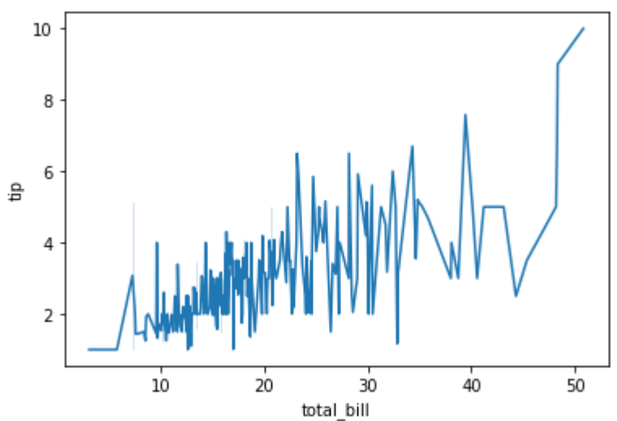
	total_bill	tip	sex	smoker	day	time	size
0	16.99	1.01	Female	No	Sun	Dinner	2
1	10.34	1.66	Male	No	Sun	Dinner	3
2	21.01	3.50	Male	No	Sun	Dinner	3
3	23.68	3.31	Male	No	Sun	Dinner	2
4	24.59	3.61	Female	No	Sun	Dinner	4
...	...	...	...	...	...	...	...
239	29.03	5.92	Male	No	Sat	Dinner	3
240	27.18	2.00	Female	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

244 rows × 7 columns

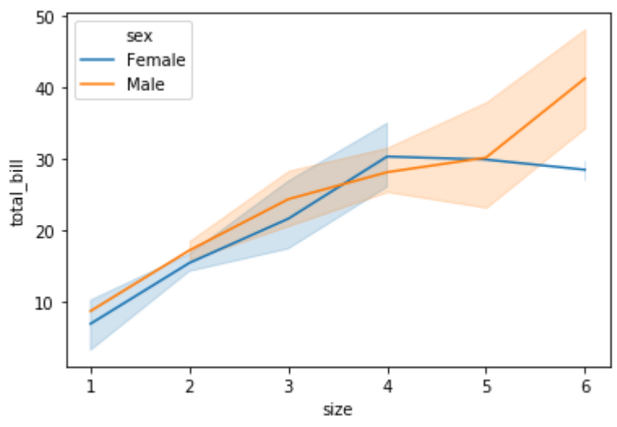
In [4]:
```
df.shape
```

Out[4]:
```
(244, 7)
```

In [5]:
```
sns.lineplot(x="total_bill",y="tip",data=df)
plt.show()
```

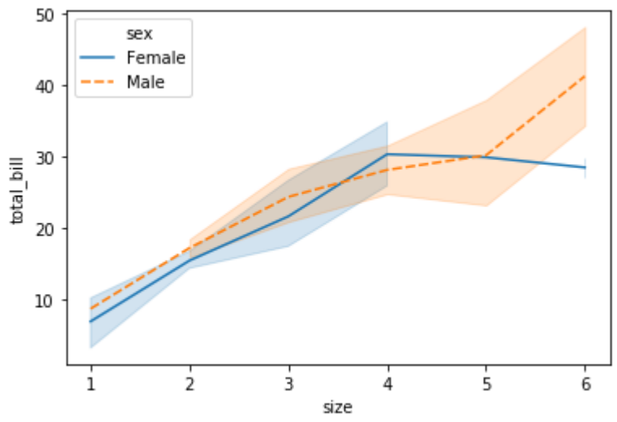


In [6]:
```
# use to hue
sns.lineplot(x="size",y="total_bill",data=df,hue="sex")
plt.show()
```



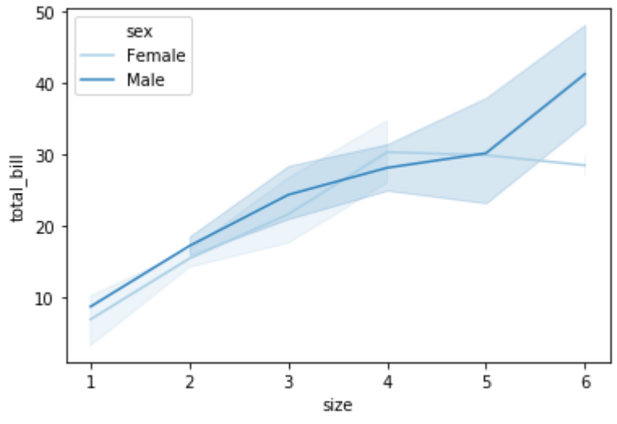
In [7]:
```
sns.lineplot(x="size",y="total_bill",data=df,hue="sex",style="sex")
```

Out[7]: <matplotlib.axes._subplots.AxesSubplot at 0x7f7b26a72850>



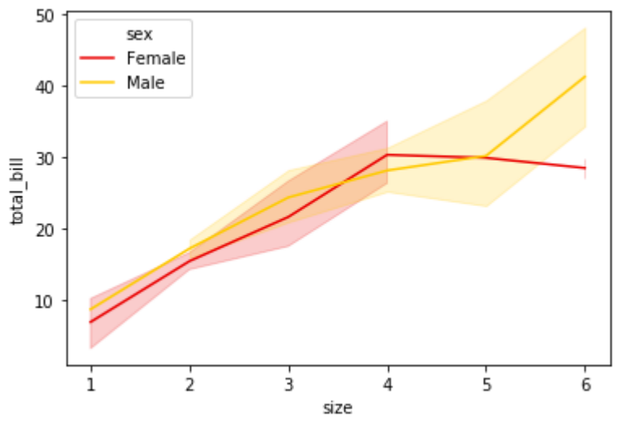
In [8]:
```
# use to palette
sns.lineplot(x="size",y="total_bill",data=df,hue="sex",palette="Blues")
```

Out[8]: <matplotlib.axes._subplots.AxesSubplot at 0x7f7b269eff90>



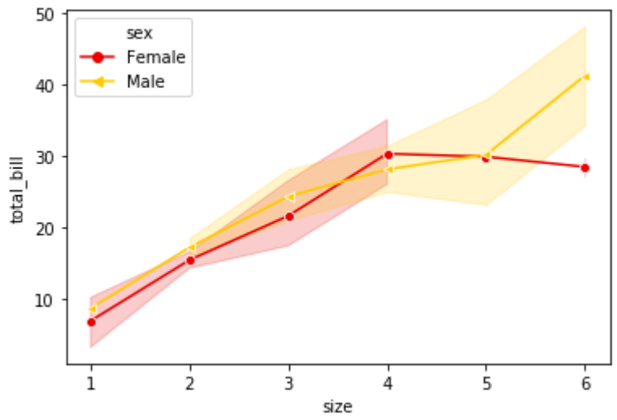
In [9]:
```
# dashes's use
sns.lineplot(x="size",y="total_bill",data=df,hue="sex",style="sex",palette="hot",dashes=False)
```

Out[9]: <matplotlib.axes._subplots.AxesSubplot at 0x7f7b26982ad0>



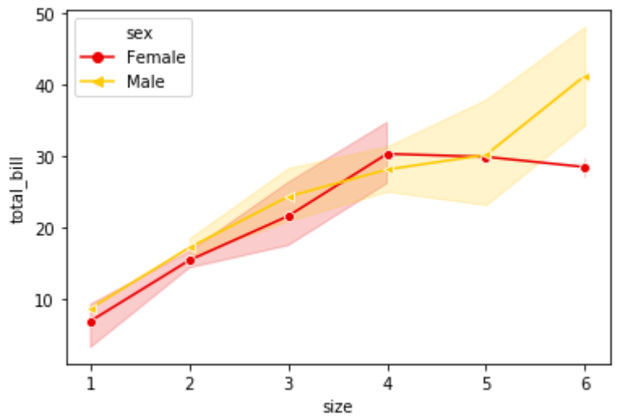
In [10]:
```
# markers's use
sns.lineplot(x="size",y="total_bill",data=df,hue="sex",style="sex",palette="hot",dashes=False,markers=["o","<"])
```

Out[10]: <matplotlib.axes._subplots.AxesSubplot at 0x7f7b2697ad50>



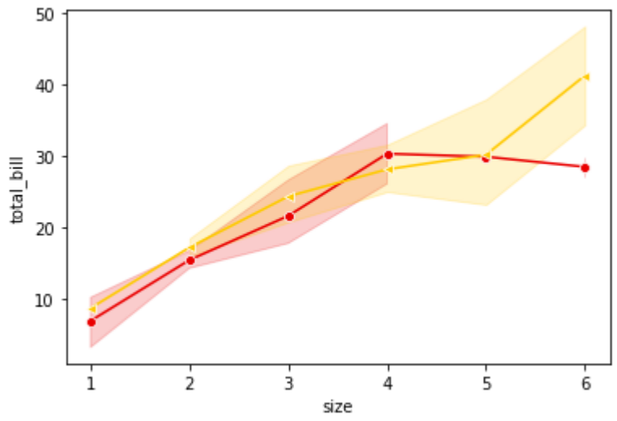
In [11]:
```
# legend's use
sns.lineplot(x="size",y="total_bill",data=df,hue="sex",style="sex",palette="hot",dashes=False,markers=["o","<"],legend="brief")
```

Out[11]: <matplotlib.axes._subplots.AxesSubplot at 0x7f7b2689b110>



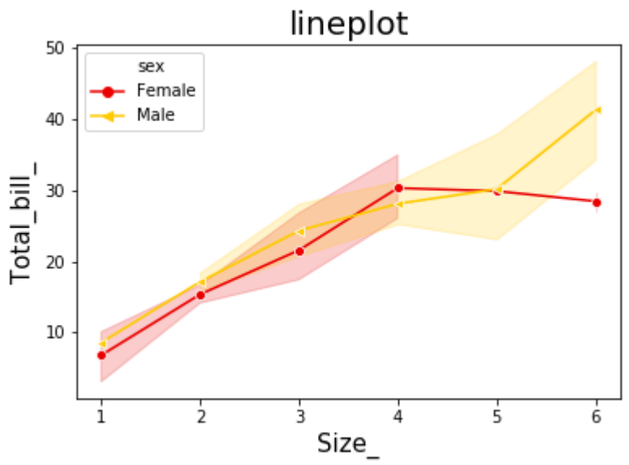
In [12]:
```
# legend's use
sns.lineplot(x="size",y="total_bill",data=df,hue="sex",style="sex",palette="hot",dashes=False,markers=["o","<"],legend=False)
```

Out[12]: <matplotlib.axes._subplots.AxesSubplot at 0x7f7b26b2d310>



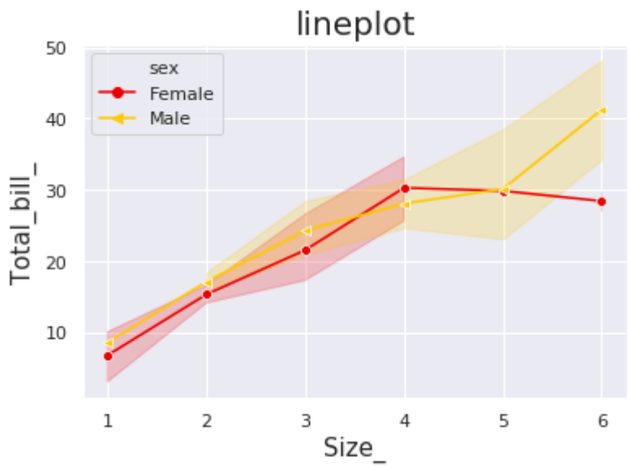
In [13]:
```
sns.lineplot(x="size",y="total_bill",data=df,hue="sex",style="sex",palette="hot",dashes=False,markers=["o","<"],legend="brief")
plt.xlabel("Size_",fontsize=15)
plt.ylabel("Total_bill_",fontsize=15)
plt.title("lineplot",fontsize=20)
```

Out[13]: Text(0.5, 1.0, 'lineplot')



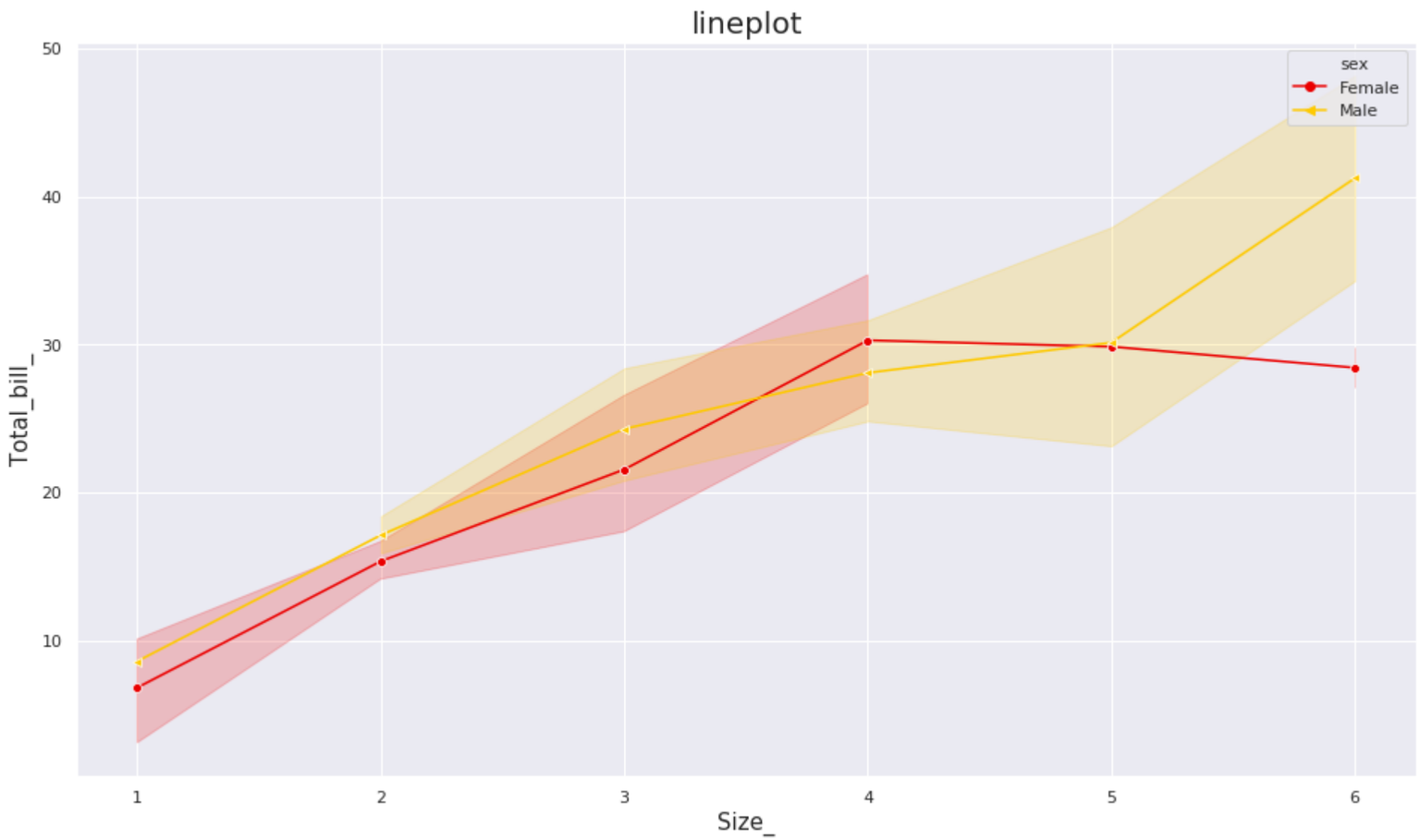
In [14]:
```
# change background color
sns.set(style="darkgrid")
sns.lineplot(x="size",y="total_bill",data=df,hue="sex",style="sex",palette="hot",dashes=False,markers=["o","<"],legend="brief")
plt.xlabel("Size_",fontsize=15)
plt.ylabel("Total_bill_",fontsize=15)
plt.title("lineplot",fontsize=20)
```

Out[14]: Text(0.5, 1.0, 'lineplot')



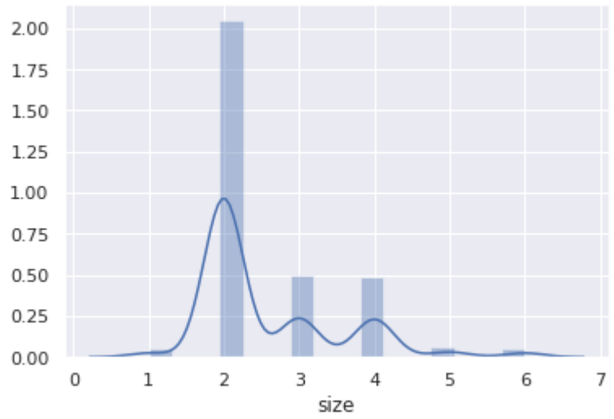
In [15]:
```
# how to increse size
plt.figure(figsize=(16,9))
sns.set(style="darkgrid")
sns.lineplot(x="size",y="total_bill",data=df,hue="sex",style="sex",palette="hot",dashes=False,markers=["o","<"],legend="brief")
plt.xlabel("Size_",fontsize=15)
plt.ylabel("Total_bill_",fontsize=15)
plt.title("lineplot",fontsize=20)
```

Out[15]: Text(0.5, 1.0, 'lineplot')



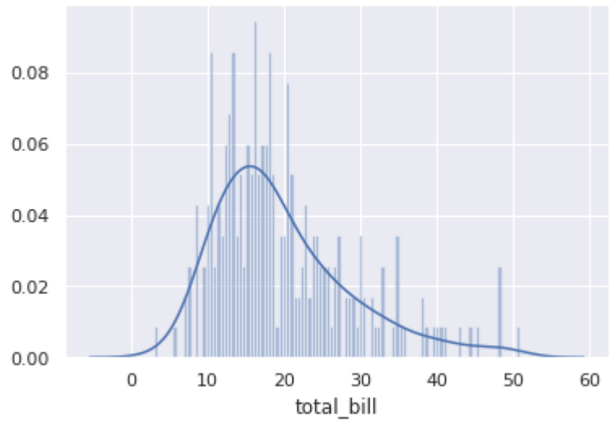
In [16]:
```
# histogram / distplot
sns.distplot(df["size"])
```

Out[16]: <matplotlib.axes._subplots.AxesSubplot at 0x7f7b2675e410>



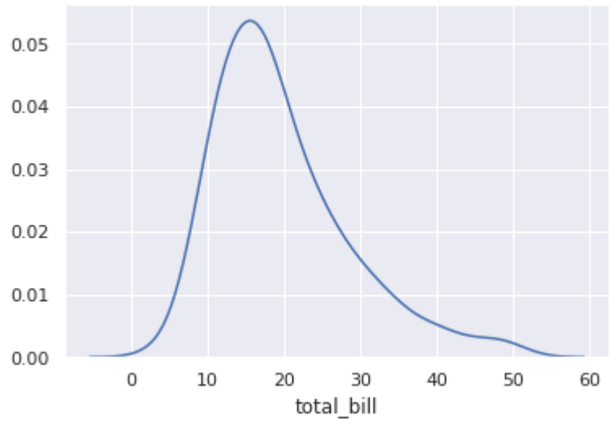
In [17]:
```
# bins's use
sns.distplot(df["total_bill"],bins=100)
```

Out[17]: <matplotlib.axes._subplots.AxesSubplot at 0x7f7b26720890>



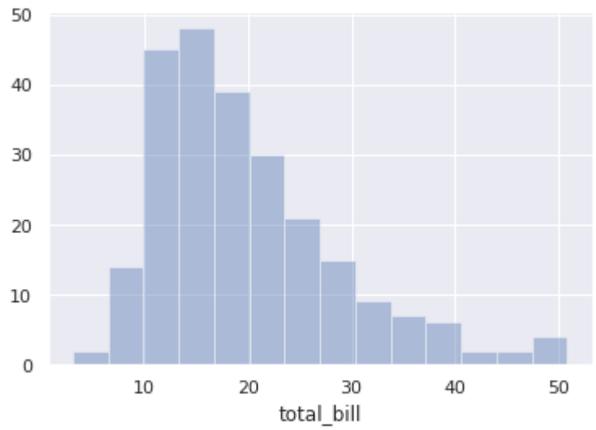
In [18]:
```
# hist's use
sns.distplot(df["total_bill"],hist=False)
```

Out[18]: <matplotlib.axes._subplots.AxesSubplot at 0x7f7b26559490>



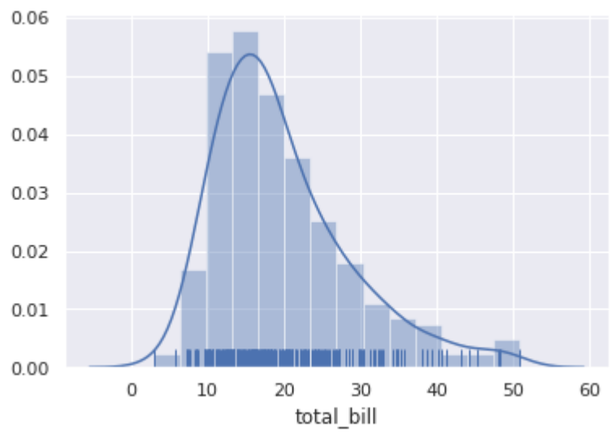
In [19]:
```
# kde's use
sns.distplot(df["total_bill"],kde=False)
```

Out[19]: <matplotlib.axes._subplots.AxesSubplot at 0x7f7b264b1d50>



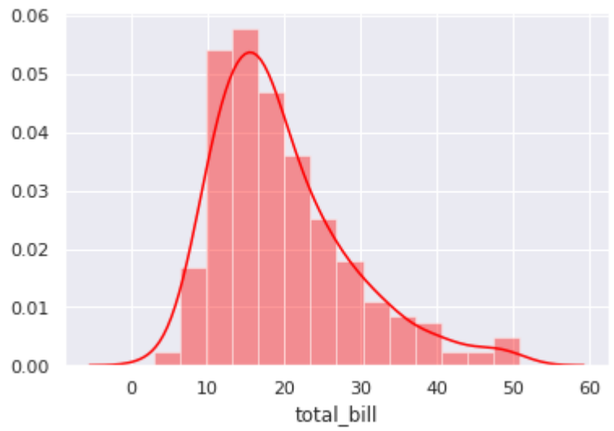
In [20]:
```
# rug's use
sns.distplot(df["total_bill"],rug=True)
```

Out[20]: <matplotlib.axes._subplots.AxesSubplot at 0x7f7b261d2190>



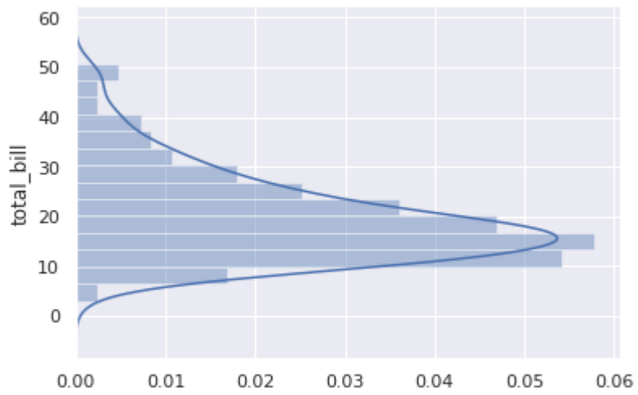
In [21]:
```
sns.distplot(df["total_bill"],color="red")
```

Out[21]: <matplotlib.axes._subplots.AxesSubplot at 0x7f7b265a2ed0>



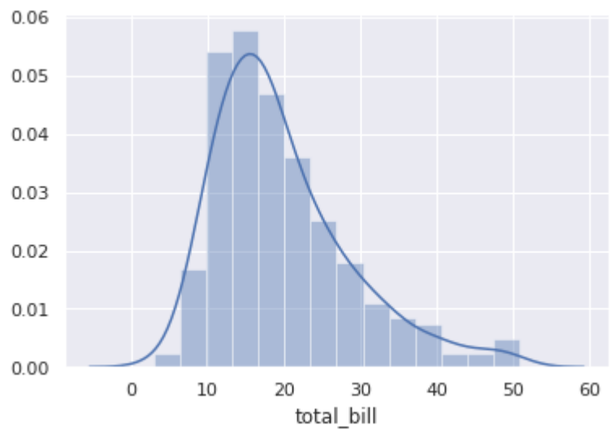
In [22]:
```
sns.distplot(df["total_bill"],vertical=True)
```

Out[22]: <matplotlib.axes._subplots.AxesSubplot at 0x7f7b26084750>



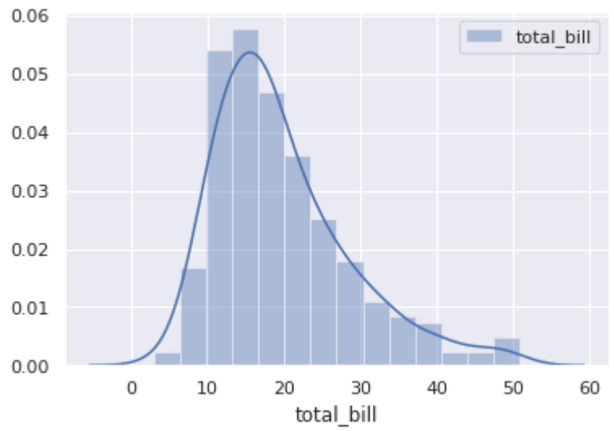
In [23]:
```
sns.distplot(df["total_bill"],axlabel="total_bill")
```

Out[23]: <matplotlib.axes._subplots.AxesSubplot at 0x7f7b26a40950>

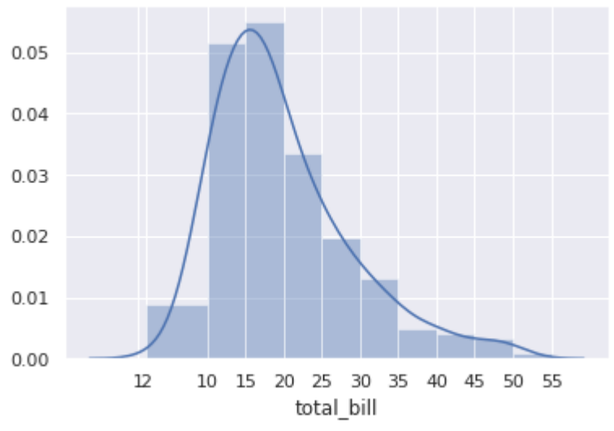


In [24]:
```
sns.distplot(df["total_bill"],label="total_bill")
plt.legend()
```

Out[24]: <matplotlib.legend.Legend at 0x7f7b26097d90>

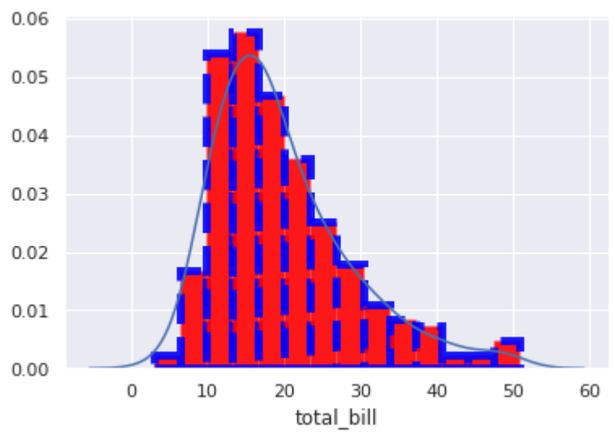


In [25]:
```
bins=[1,2,10,15,20,25,30,35,40,45,50,55]
sns.distplot(df["total_bill"],bins=bins)
plt.xticks(bins)
plt.show()
```



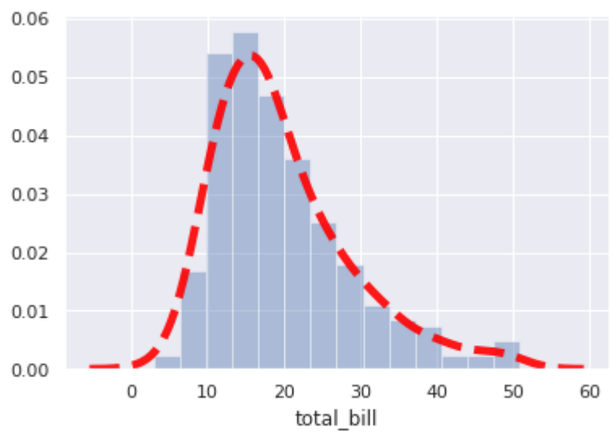
In [26]:
```
# hist_kws
sns.distplot(df["total_bill"],hist_kws={
    "color":"red",
    "edgecolor":"blue",
    "linewidth":"5",
    "linestyle":"--",
    "alpha":0.9,

})
```

Out[26]: <matplotlib.axes._subplots.AxesSubplot at 0x7f7b2679c190>



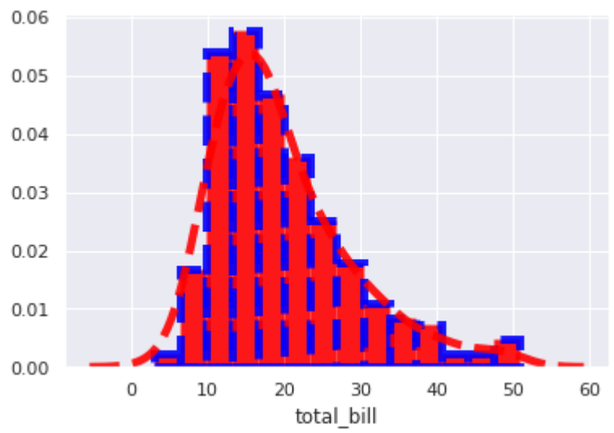
In [27]:
```
# kde_kws
sns.distplot(df["total_bill"],kde_kws={
    "color":"red",
    "linewidth":"5",
    "linestyle":"--",
    "alpha":0.9,

})
```

Out[27]: <matplotlib.axes._subplots.AxesSubplot at 0x7f7b25ebb490>



In [28]:
```
sns.distplot(df["total_bill"],hist_kws={
    "color":"red",
    "edgecolor":"blue",
    "linewidth":"5",
    "linestyle":"--",
    "alpha":0.9,

},kde_kws={
    "color":"red",
    "linewidth":"5",
    "linestyle":"--",
    "alpha":0.9,}

            )
```

Out[28]:
```
<matplotlib.axes._subplots.AxesSubplot at 0x7f7b25d16d90>
```



barplot

In [29]:
```
br=sns.load_dataset("tips")
br
```

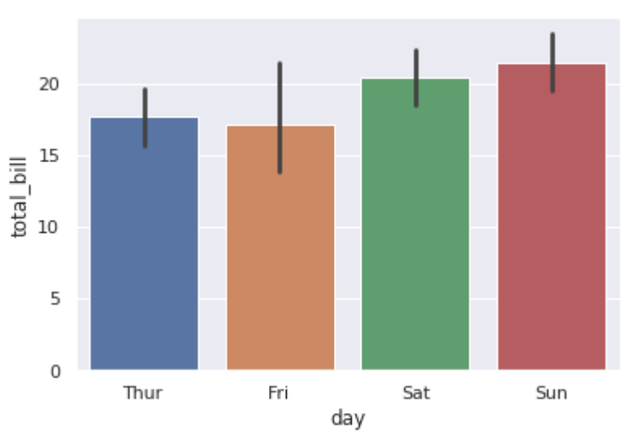
Out[29]:

	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

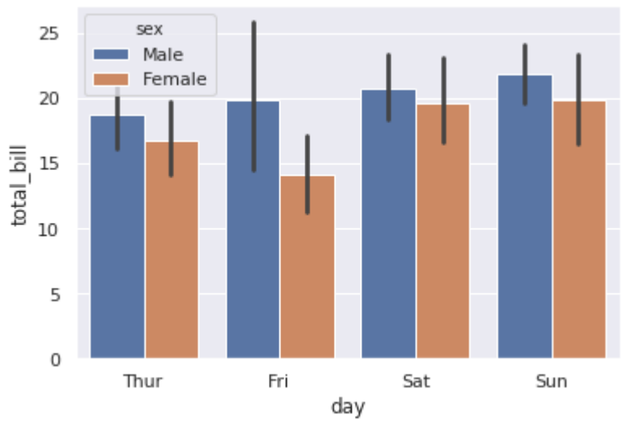
In [30]:
```
sns.barplot(x="day",y="total_bill",data=br)
```

Out[30]:
```
<matplotlib.axes._subplots.AxesSubplot at 0x7f7b25cff050>
```



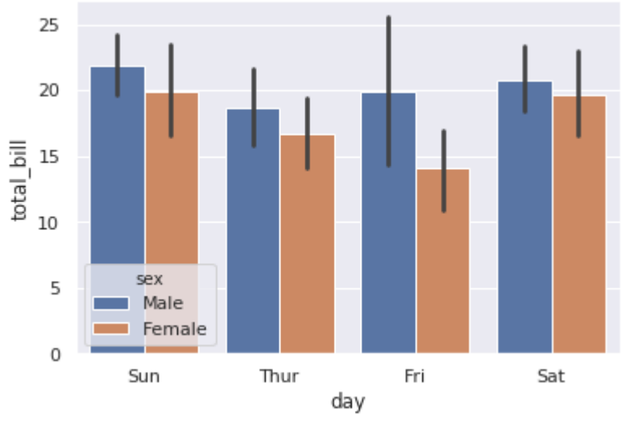
In [31]:
```
sns.barplot(x="day",y="total_bill",data=br,hue="sex")
```

Out[31]: <matplotlib.axes._subplots.AxesSubplot at 0x7f7b25c63450>



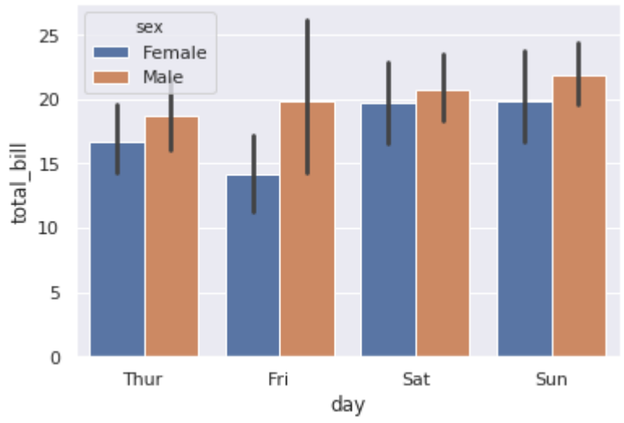
In [32]:
```
# order
order=["Sun","Thur","Fri","Sat"]
sns.barplot(x="day",y="total_bill",data=br,hue="sex",order=order)
```

Out[32]: <matplotlib.axes._subplots.AxesSubplot at 0x7f7b25c07810>



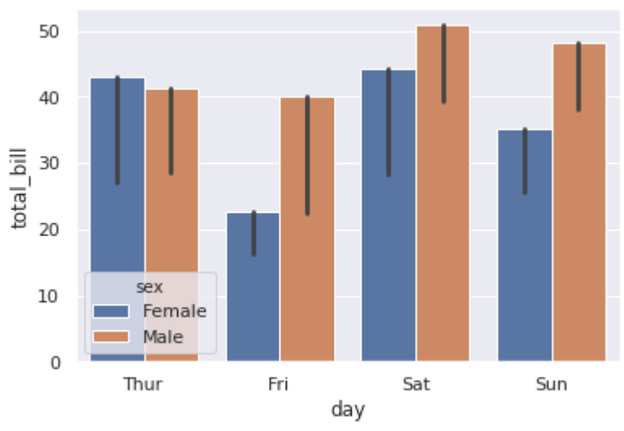
In [33]:
```
sns.barplot(x="day",y="total_bill",data=br,hue="sex",hue_order=["Female","Male"])
```

Out[33]: <matplotlib.axes._subplots.AxesSubplot at 0x7f7b25b65110>



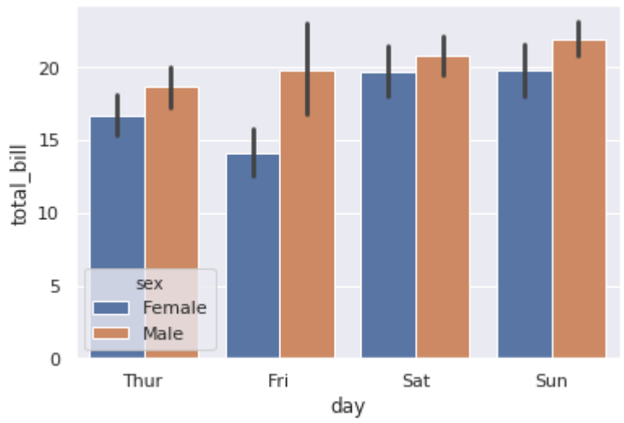
In [34]:
```
# y axis values change

sns.barplot(x="day",y="total_bill",data=br,hue="sex",hue_order=["Female","Male"],estimator=np.max)
```

Out[34]: <matplotlib.axes._subplots.AxesSubplot at 0x7f7b25ab66d0>



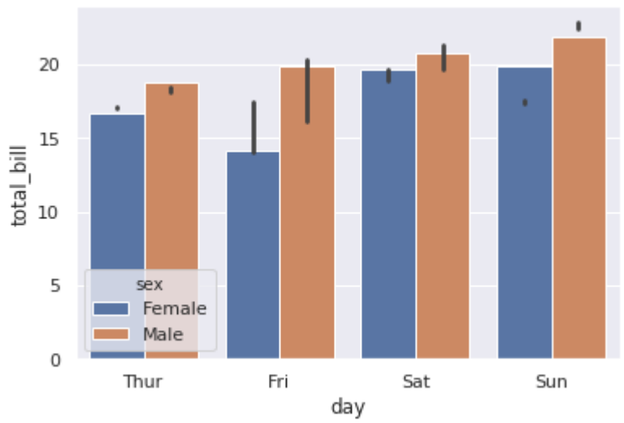
In [35]:
```
sns.barplot(x="day",y="total_bill",data=br,hue="sex",hue_order=["Female","Male"],ci=70
           )
```

Out[35]: <matplotlib.axes._subplots.AxesSubplot at 0x7f7b25a80d10>



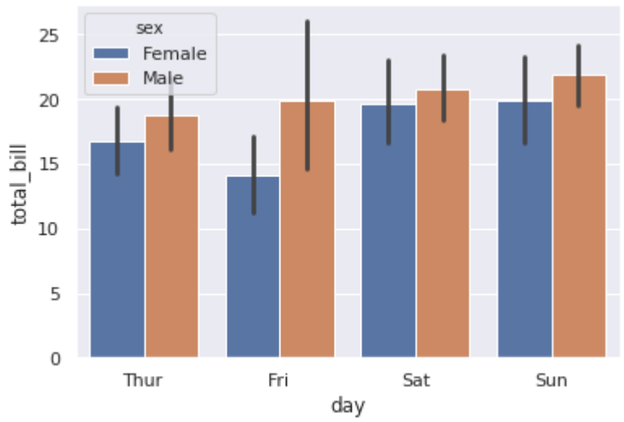
In [36]:
```
sns.barplot(x="day",y="total_bill",data=br,hue="sex",hue_order=["Female","Male"],n_boot=2)
```

Out[36]: <matplotlib.axes._subplots.AxesSubplot at 0x7f7b25e9db50>



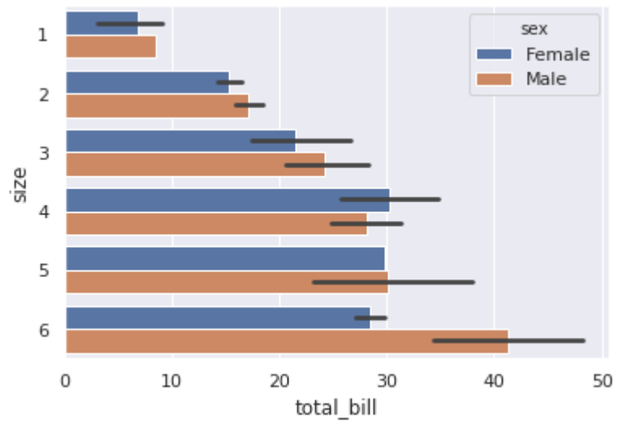
In [37]:
```
sns.barplot(x="day",y="total_bill",data=br,hue="sex",hue_order=["Female","Male"],orient="v")
```

Out[37]: <matplotlib.axes._subplots.AxesSubplot at 0x7f7b2594e450>



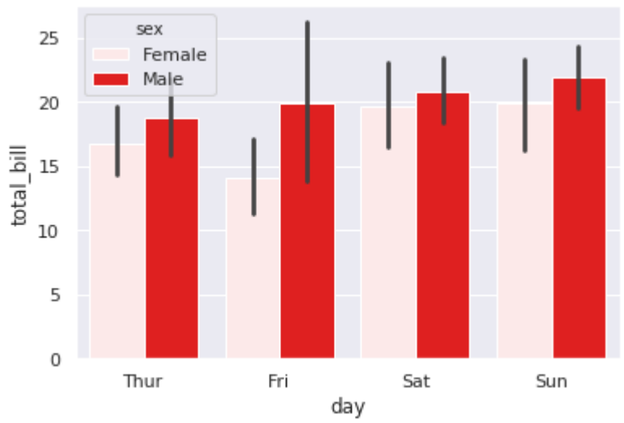
In [38]:
```
sns.barplot(x="total_bill",y="size",data=br,hue="sex",hue_order=["Female","Male"],orient="h")
```

Out[38]: <matplotlib.axes._subplots.AxesSubplot at 0x7f7b25912610>



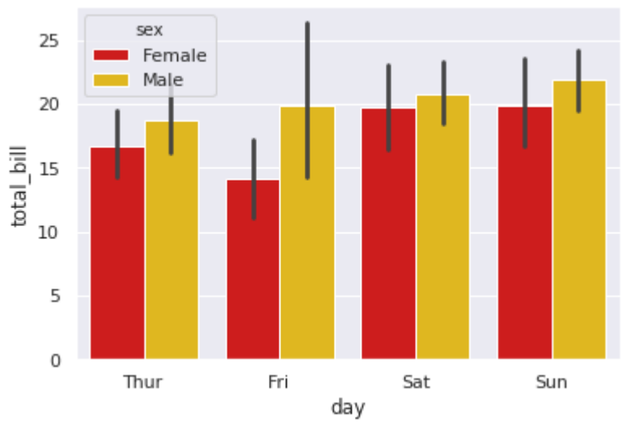
In [39]:
```
sns.barplot(x="day",y="total_bill",data=br,hue="sex",hue_order=["Female","Male"],color="red")
```

Out[39]: <matplotlib.axes._subplots.AxesSubplot at 0x7f7b2581bdd0>



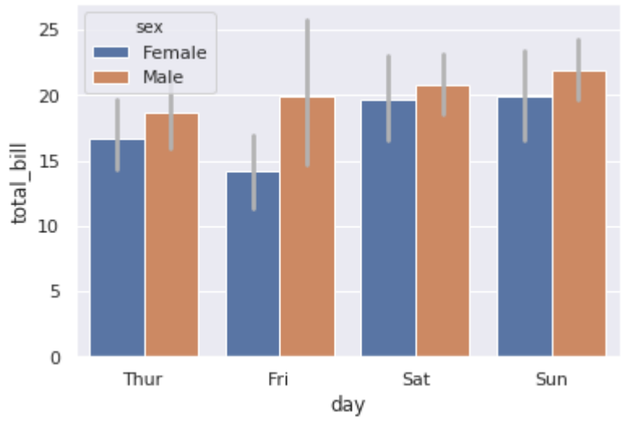
In [40]:
```
sns.barplot(x="day",y="total_bill",data=br,hue="sex",hue_order=["Female","Male"],palette="hot")
```

Out[40]: <matplotlib.axes._subplots.AxesSubplot at 0x7f7b2579a150>



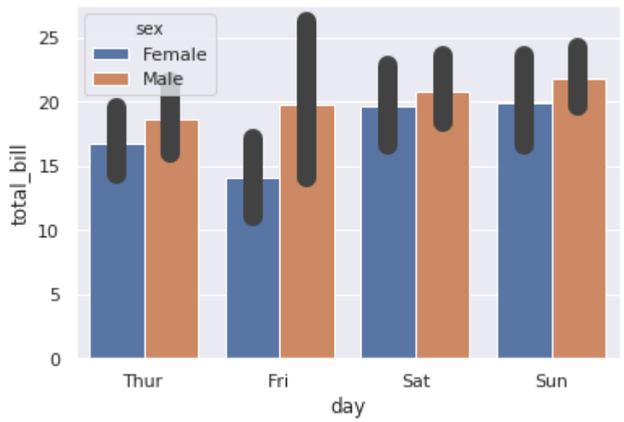
In [41]:
```
sns.barplot(x="day",y="total_bill",data=br,hue="sex",hue_order=["Female","Male"],errcolor="0.7")
```

Out[41]: <matplotlib.axes._subplots.AxesSubplot at 0x7f7b2571e590>



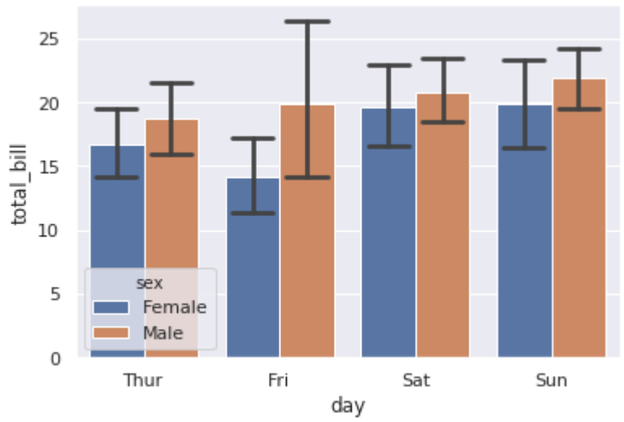
In [42]:
```
sns.barplot(x="day",y="total_bill",data=br,hue="sex",hue_order=["Female","Male"],errwidth=12)
```

Out[42]: <matplotlib.axes._subplots.AxesSubplot at 0x7f7b2571ef90>



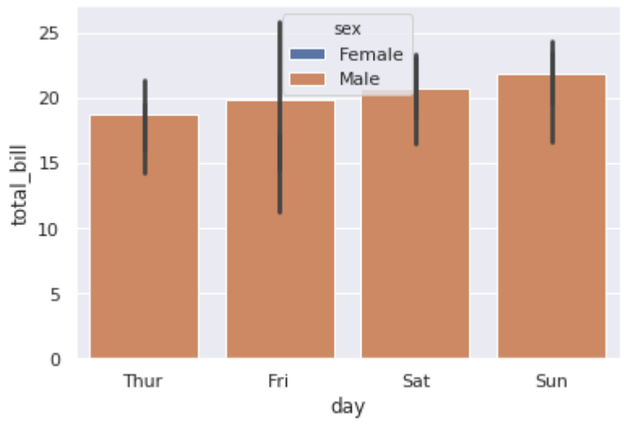
In [43]:
```
sns.barplot(x="day",y="total_bill",data=br,hue="sex",hue_order=["Female","Male"],capsize=0.3)
```

Out[43]: <matplotlib.axes._subplots.AxesSubplot at 0x7f7b255c6b90>



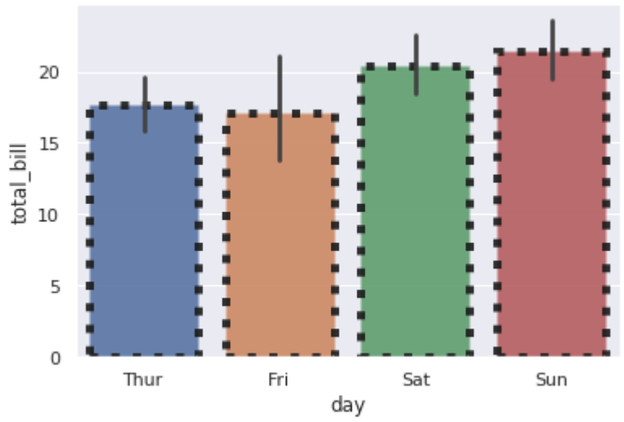
In [44]:
```
sns.barplot(x="day",y="total_bill",data=br,hue="sex",hue_order=["Female","Male"],dodge=False)
```

Out[44]: <matplotlib.axes._subplots.AxesSubplot at 0x7f7b255078d0>



In [45]:
```
sns.set()
kwargs=kwargs={
    'alpha':0.9,
    'linestyle':':',
    'linewidth':5,
    'edgecolor':'k'}
sns.barplot(x="day",y="total_bill",data=br,**kwargs)
```

Out[45]: <matplotlib.axes._subplots.AxesSubplot at 0x7f7b254a3910>



scatter plot

In [46]:
```
df=sns.load_dataset("titanic")
df
```

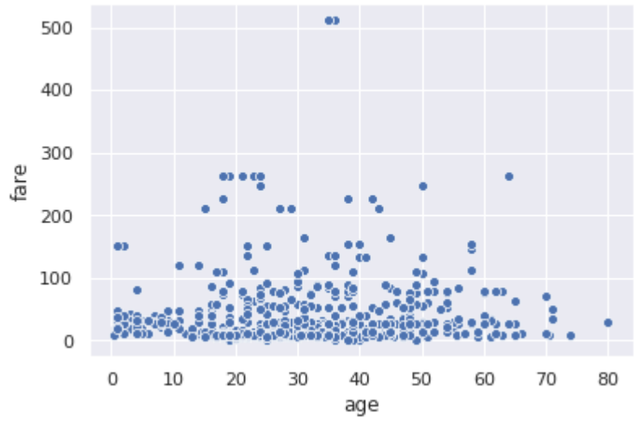
Out[46]:

	survived	pclass	sex	age	sibsp	parch	fare	embarked	class	who	adult_male	deck	embark_town	alive	alone
0	0	3	male	22.0	1	0	7.2500	S	Third	man	True	NaN	Southampton	no	False
1	1	1	female	38.0	1	0	71.2833	C	First	woman	False	C	Cherbourg	yes	False
2	1	3	female	26.0	0	0	7.9250	S	Third	woman	False	NaN	Southampton	yes	True
3	1	1	female	35.0	1	0	53.1000	S	First	woman	False	C	Southampton	yes	False
4	0	3	male	35.0	0	0	8.0500	S	Third	man	True	NaN	Southampton	no	True
...	...	...	...	...	...	...	...	...	...	...	...	...	...	...	...
886	0	2	male	27.0	0	0	13.0000	S	Second	man	True	NaN	Southampton	no	True
887	1	1	female	19.0	0	0	30.0000	S	First	woman	False	B	Southampton	yes	True
888	0	3	female	NaN	1	2	23.4500	S	Third	woman	False	NaN	Southampton	no	False
889	1	1	male	26.0	0	0	30.0000	C	First	man	True	C	Cherbourg	yes	True
890	0	3	male	32.0	0	0	7.7500	Q	Third	man	True	NaN	Queenstown	no	True

891 rows × 15 columns

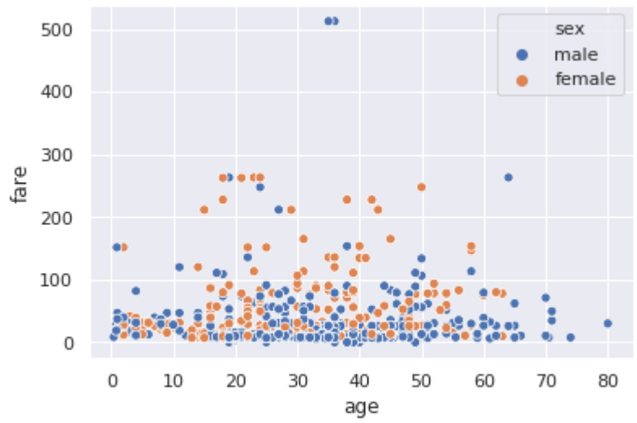
In [47]:
```
sns.scatterplot(x="age",y="fare",data=df)
```

Out[47]: <matplotlib.axes._subplots.AxesSubplot at 0x7f7b254132d0>



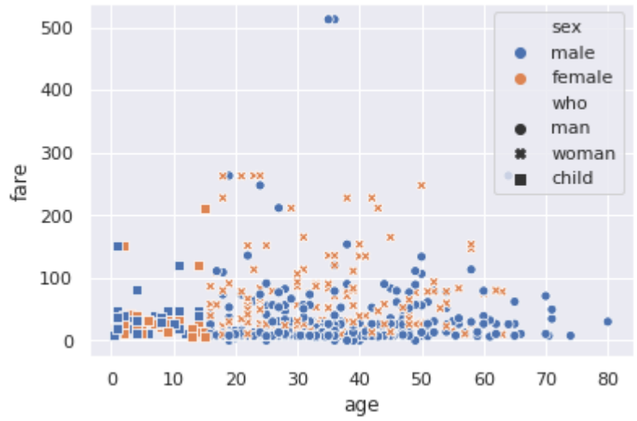
In [48]:
```
sns.scatterplot(x="age",y="fare",data=df,hue="sex")
```

Out[48]: <matplotlib.axes._subplots.AxesSubplot at 0x7f7b253a66d0>



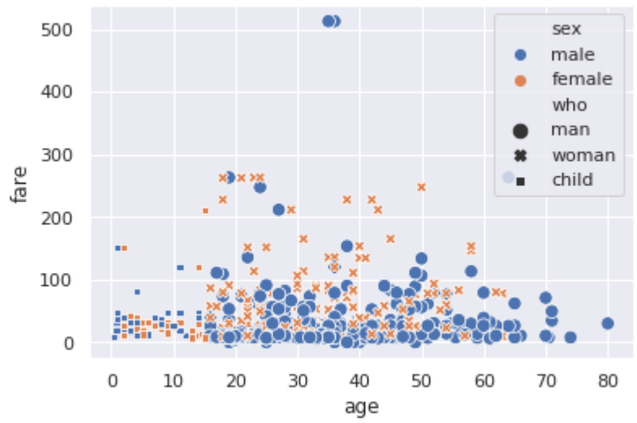
In [49]:
```
sns.scatterplot(x="age",y="fare",data=df,hue="sex",style="who")
```

Out[49]: <matplotlib.axes._subplots.AxesSubplot at 0x7f7b25392c50>



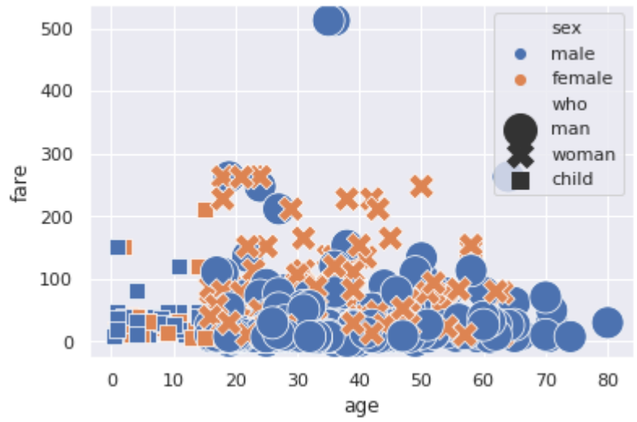
In [50]:
```
sns.scatterplot(x="age",y="fare",data=df,hue="sex",style="who",size="who")
```

Out[50]: <matplotlib.axes._subplots.AxesSubplot at 0x7f7b25331f10>



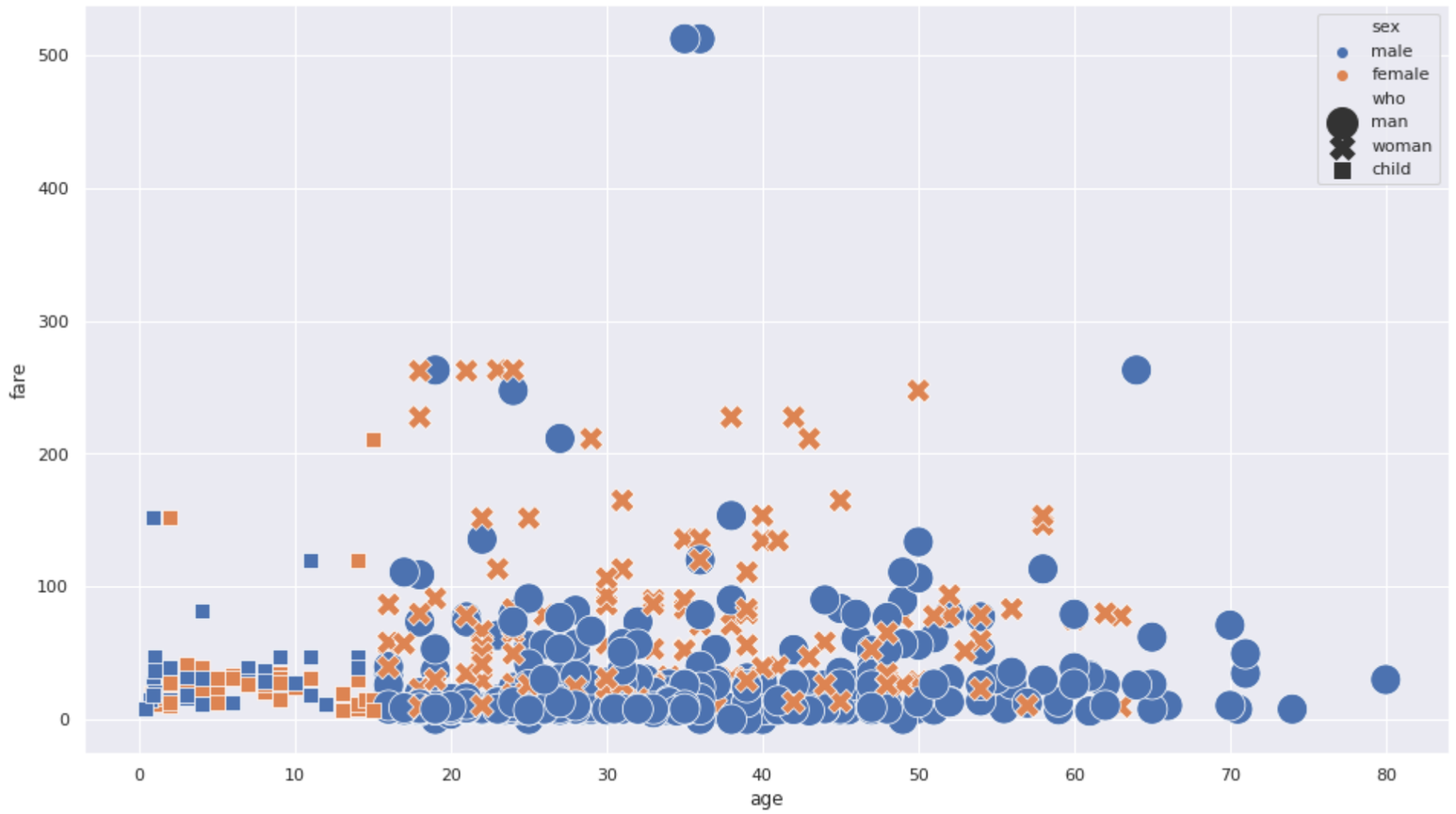
In [51]:
```
sns.scatterplot(x="age",y="fare",data=df,hue="sex",style="who",size="who",sizes=(100,400))
```

Out[51]: <matplotlib.axes._subplots.AxesSubplot at 0x7f7b2524ed50>



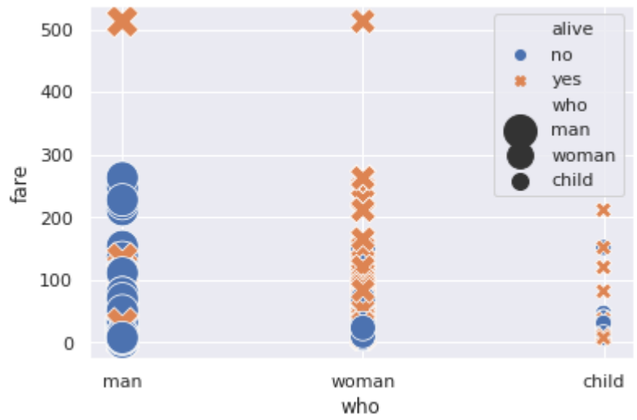
In [52]:
```
plt.figure(figsize=(16,9))
sns.scatterplot(x="age",y="fare",data=df,hue="sex",style="who",size="who",sizes=(100,400))
```

Out[52]: <matplotlib.axes._subplots.AxesSubplot at 0x7f7b25106d90>



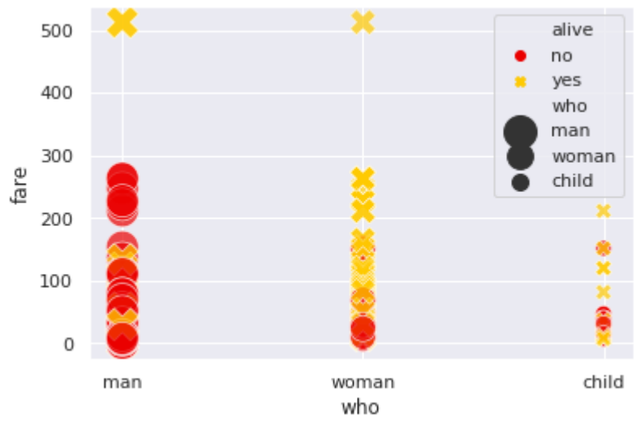
In [53]:
```
sns.scatterplot(x="who",y="fare",data=df,hue="alive",style="alive",size="who",sizes=(100,400))
```

Out[53]: <matplotlib.axes._subplots.AxesSubplot at 0x7f7b250a0110>



In [54]:
```
sns.scatterplot(x="who",y="fare",data=df,hue="alive",style="alive",size="who",sizes=(100,400),palette="hot",alpha=0.7)
```

Out[54]: <matplotlib.axes._subplots.AxesSubplot at 0x7f7b2400a990>



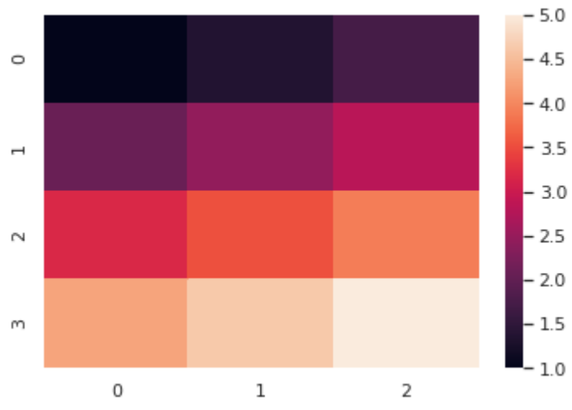
heatmap

In [55]:
```
arr_2d=np.linspace(1,5,12).reshape(4,3)
arr_2d
```

Out[55]:
```
array([[1.        , 1.36363636, 1.72727273],
       [2.09090909, 2.45454545, 2.81818182],
       [3.18181818, 3.54545455, 3.90909091],
       [4.27272727, 4.63636364, 5.        ]])
```

In [56]:
```
sns.heatmap(arr_2d)
```

Out[56]:
```
<matplotlib.axes._subplots.AxesSubplot at 0x7f7b1f7cd190>
```



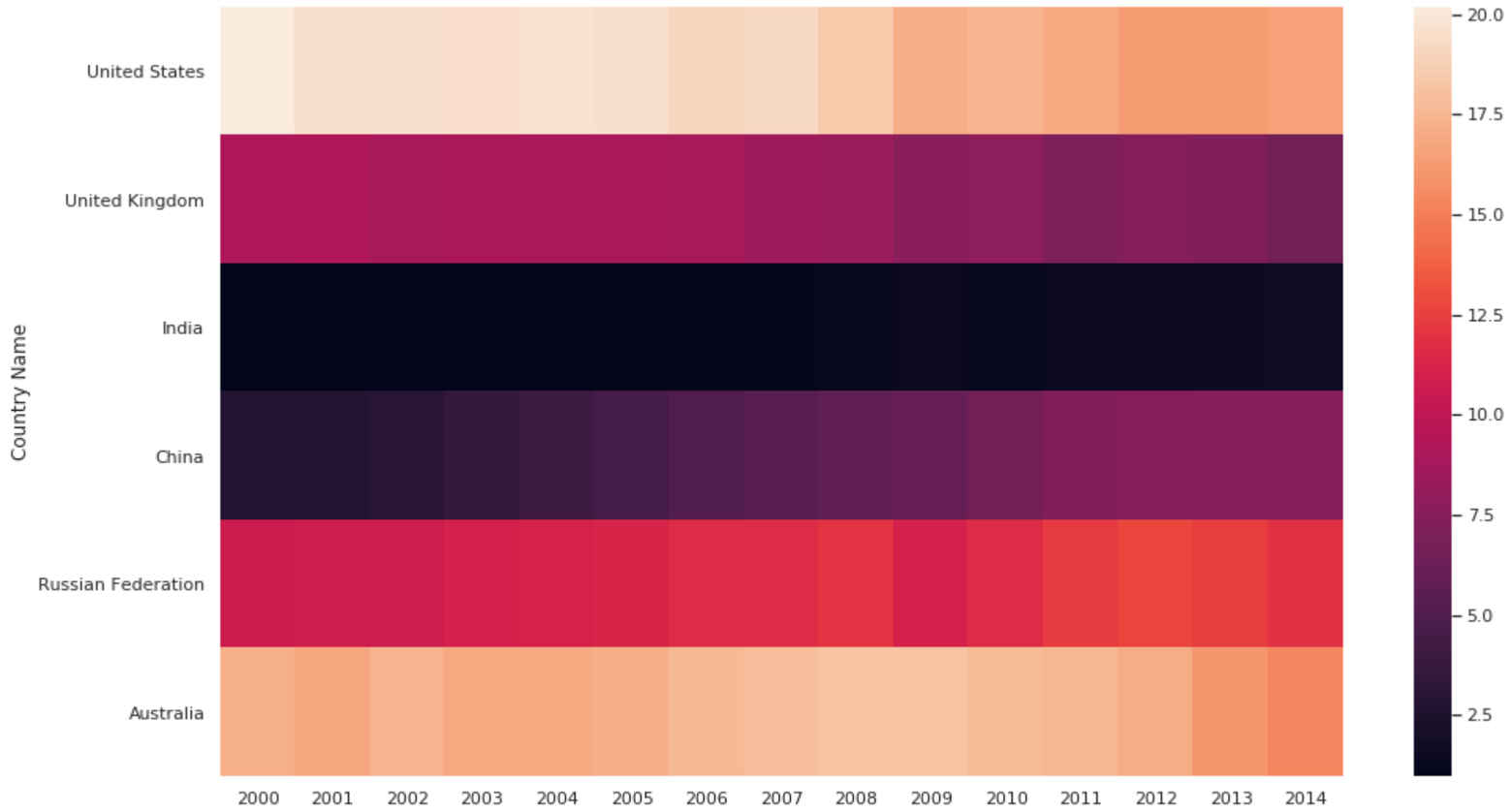
In [57]:
```
hf=pd.read_csv("Who_is_responsible_for_global_warming.csv")
hf=hf.drop(["Country Code","Indicator Name","Indicator Code"],axis=1).set_index("Country Name")
hf=hf.head(6)
hf
```

Out[57]:

Country Name	2000	2001	2002	2003	2004	2005	2006	2007	2008	2009	2010	2011	2012	2013	2014
United States	20.178751	19.636505	19.613404	19.564105	19.658371	19.591885	19.094067	19.217898	18.461764	17.157738	17.442862	16.976957	16.310471	16.323477	16.502837
United Kingdom	9.199549	9.233175	8.904123	9.053278	8.989140	8.982939	8.898710	8.617164	8.424424	7.574622	7.857836	7.079298	7.355898	7.145844	6.497440
India	0.979870	0.971698	0.967381	0.992392	1.025028	1.068563	1.121982	1.193210	1.310098	1.431844	1.397009	1.476686	1.598099	1.591438	1.730000
China	2.696862	2.742121	3.007083	3.524074	4.037991	4.523178	4.980314	5.334910	5.701915	6.010102	6.560520	7.241515	7.424751	7.557211	7.543908
Russian Federation	10.627121	10.669603	10.715901	11.090647	11.120627	11.253529	11.669122	11.672457	12.014507	11.023856	11.694348	12.334881	12.784979	12.393556	11.857528
Australia	17.200610	16.733367	17.370452	16.901959	17.026515	17.169711	17.651398	17.865260	18.160876	18.200182	17.740845	17.538878	17.072905	16.095833	15.388766

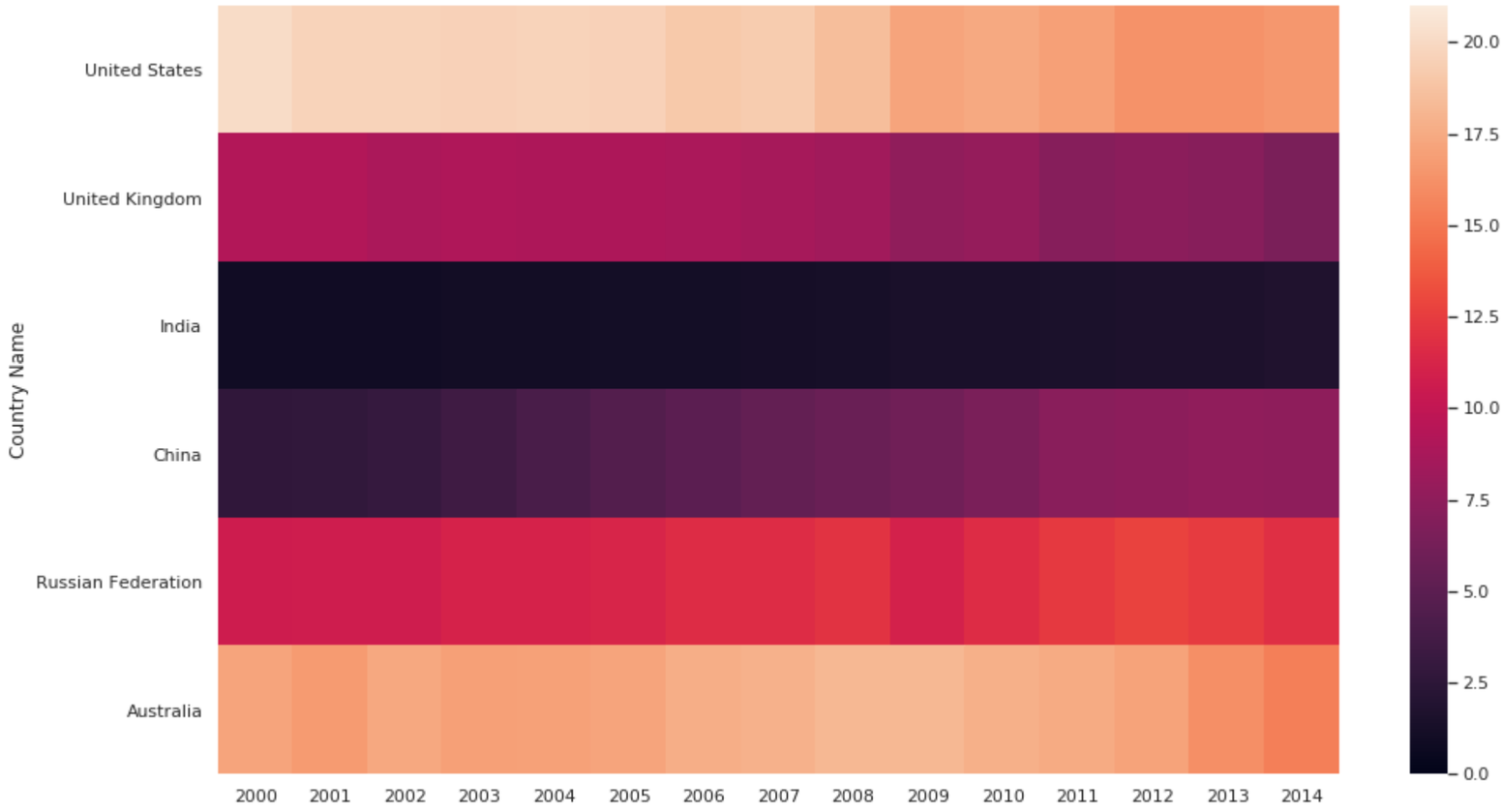
In [58]:
```
plt.figure(figsize=(16,9))
sns.heatmap(hf)
```

Out[58]: <matplotlib.axes._subplots.AxesSubplot at 0x7f7b1e67e410>



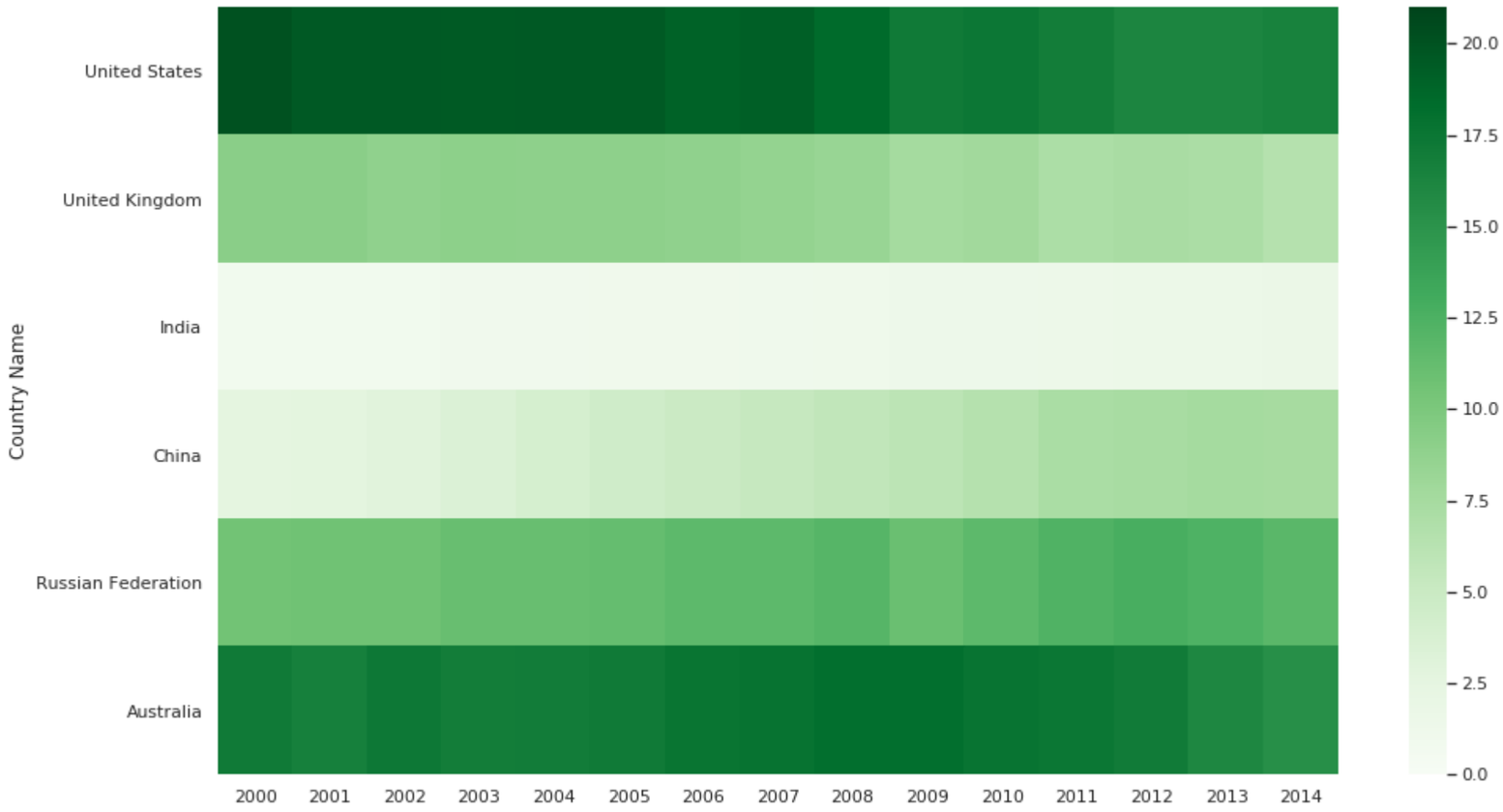
In [59]:
```
# set min,max value of sidebar
plt.figure(figsize=(16,9))
sns.heatmap(hf,vmin=0,vmax=21)
```

Out[59]: <matplotlib.axes._subplots.AxesSubplot at 0x7f7b1e69fcd0>



In [60]:
```
plt.figure(figsize=(16,9))
sns.heatmap(hf,vmin=0,vmax=21,cmap="Greens")
```

Out[60]: <matplotlib.axes._subplots.AxesSubplot at 0x7f7b1f78e950>

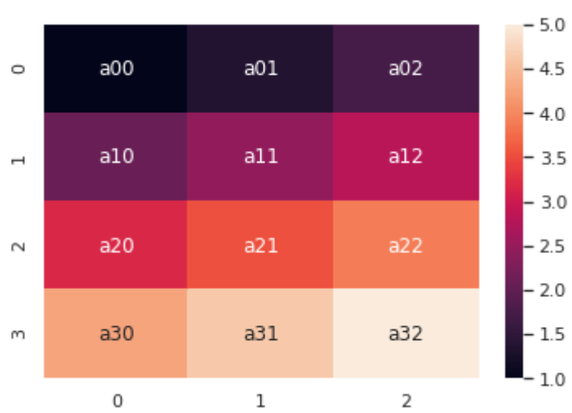


In [61]:
```
plt.figure(figsize=(16,9))
sns.heatmap(hf,vmin=0,vmax=21,cmap="Greens",annot=True)
```

Out[61]: <matplotlib.axes._subplots.AxesSubplot at 0x7f7b1e40b7d0>

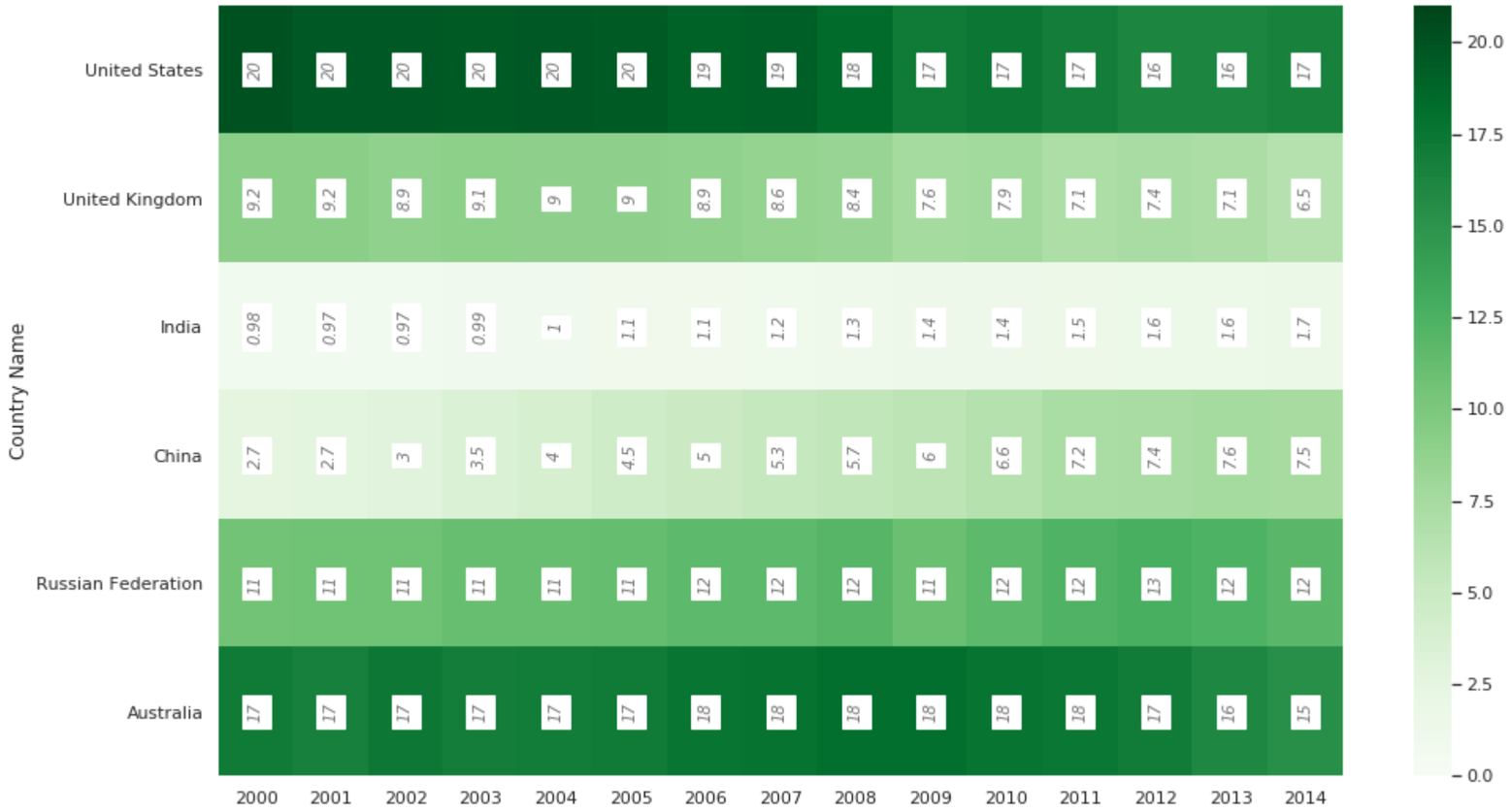
In [62]:
```
# how to set default annot
# fmt is type of datatype here we use string
annot_arr=np.array([["a00","a01","a02"],["a10","a11","a12"],["a20","a21","a22"],["a30","a31","a32"]])
sns.heatmap(arr_2d,annot=annot_arr,fmt="s")
```

Out[62]: <matplotlib.axes._subplots.AxesSubplot at 0x7f7b1e2eab50>



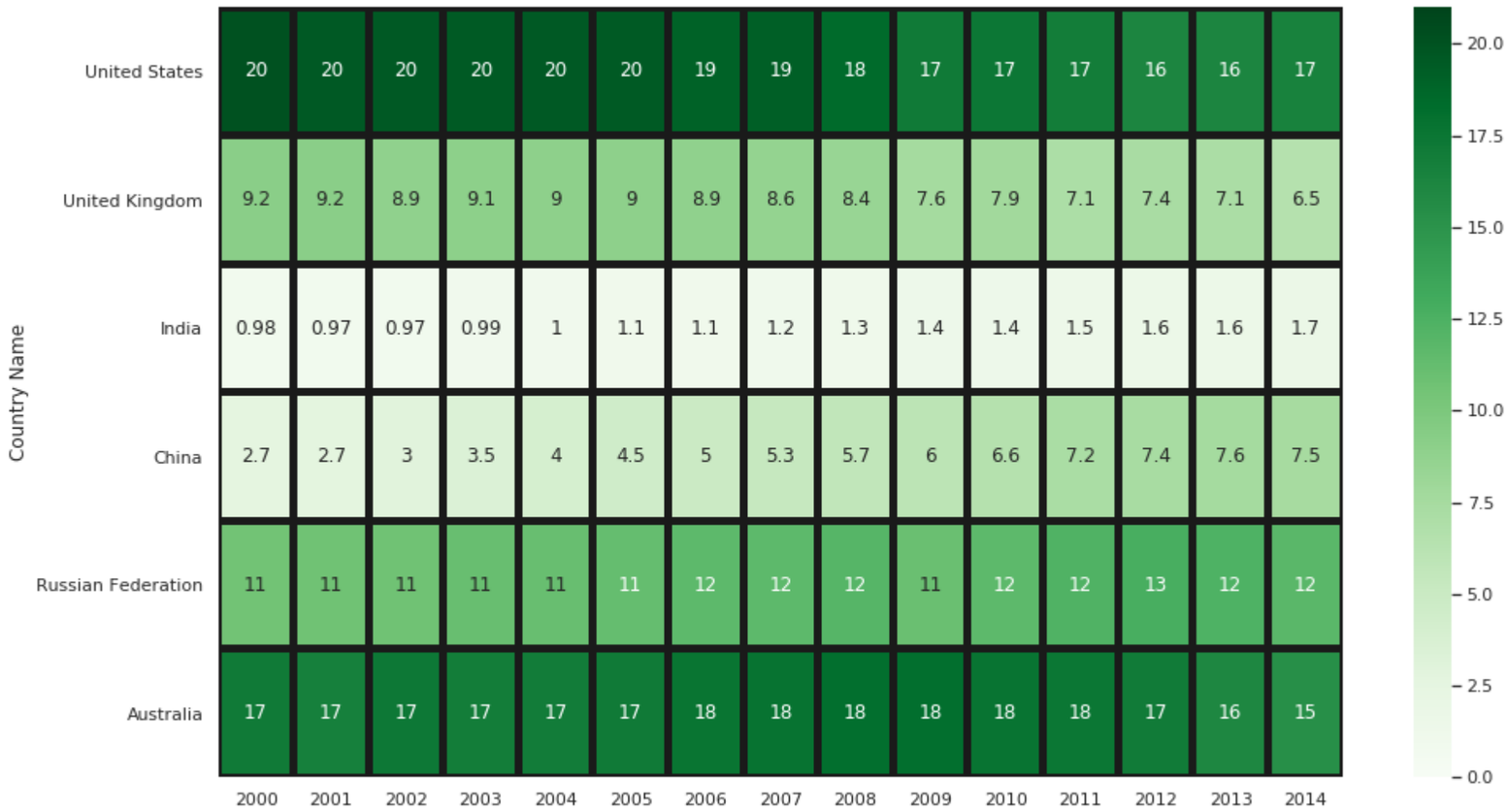
In [63]:
```
# annot_kws
plt.figure(figsize=(16,9))
annot_kws={
    "fontsize":10,
    "fontstyle":"italic",
    "color":"k",
    "alpha":0.6,
    "rotation":"vertical",
    "verticalalignment":"center",
    "backgroundcolor":"w"
}
sns.heatmap(hf,vmin=0,vmax=21,cmap="Greens",annot=True,annot_kws=annot_kws)
```

Out[63]: <matplotlib.axes._subplots.AxesSubplot at 0x7f7b1e204390>



In [64]:
```
plt.figure(figsize=(16,9))
sns.heatmap(hf,vmin=0,vmax=21,cmap="Greens",annot=True,linewidth=4,linecolor="k")
```

Out[64]: <matplotlib.axes._subplots.AxesSubplot at 0x7f7b1e13e490>

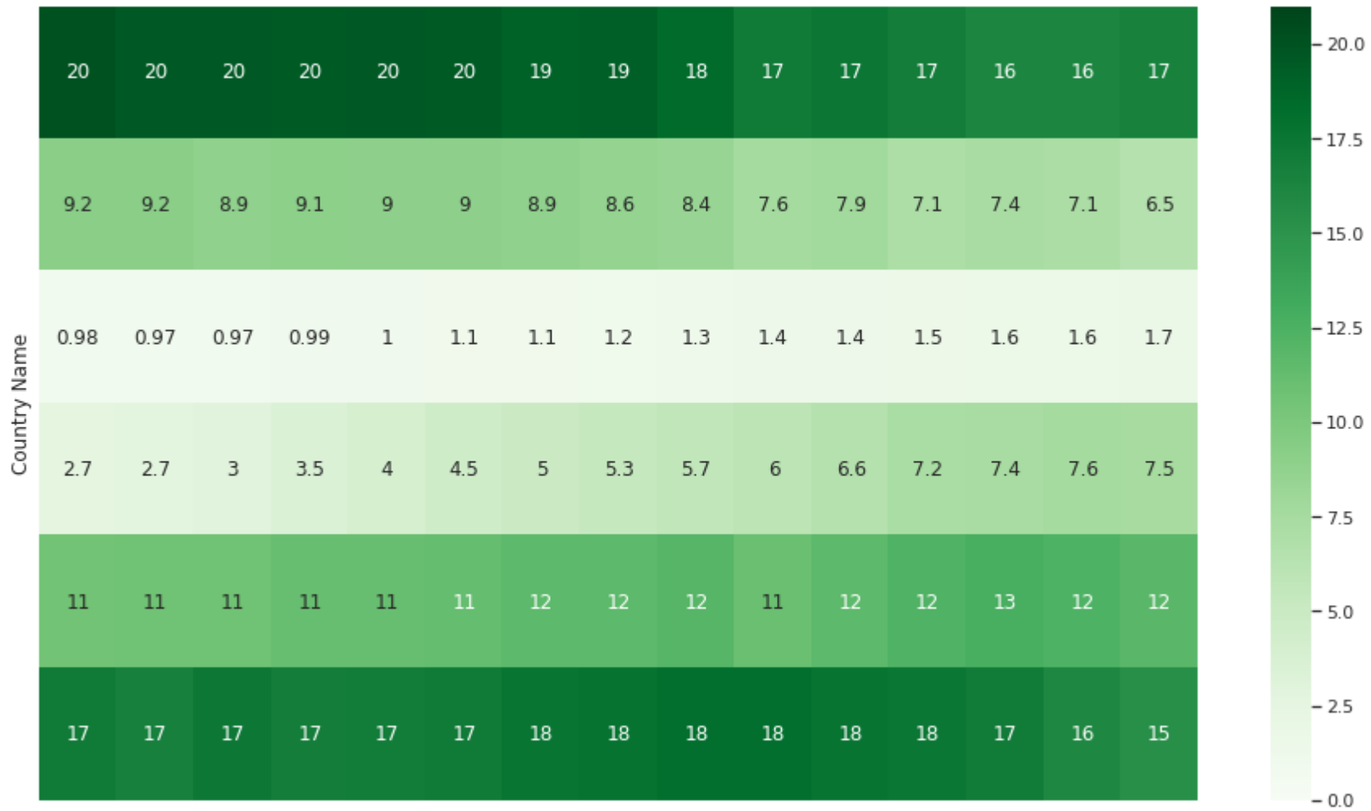


In [65]:
```
# hide cbar
plt.figure(figsize=(16,9))
sns.heatmap(hf,vmin=0,vmax=21,cmap="Greens",annot=True,cbar=False)
```

Out[65]: <matplotlib.axes._subplots.AxesSubplot at 0x7f7b1dfd60d0>

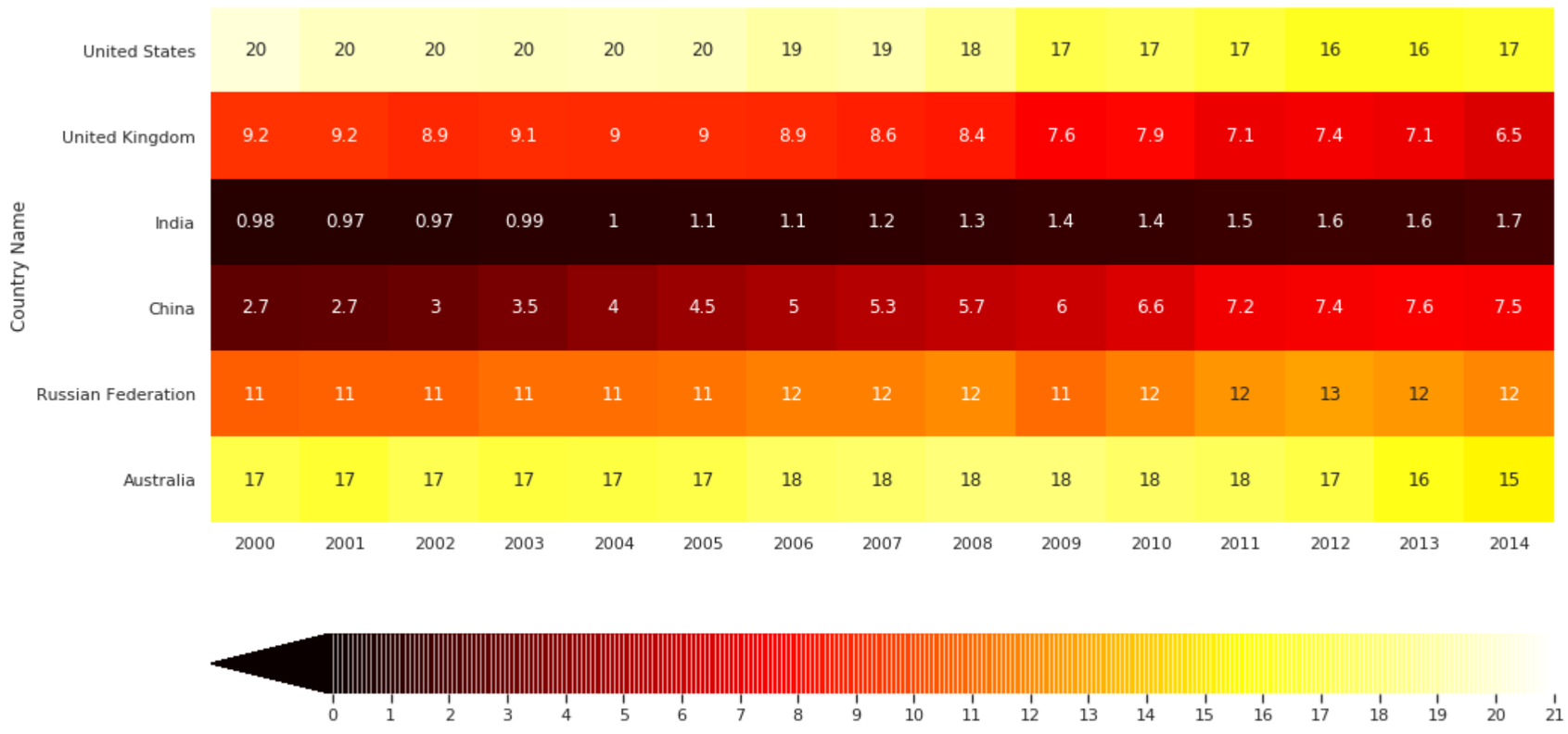
In [66]:
```
# remove x and ticks
plt.figure(figsize=(16,9))
sns.heatmap(hf,vmin=0,vmax=21,cmap="Greens",annot=True,xticklabels=False,yticklabels=False)
```

Out[66]: <matplotlib.axes._subplots.AxesSubplot at 0x7f7b1de852d0>

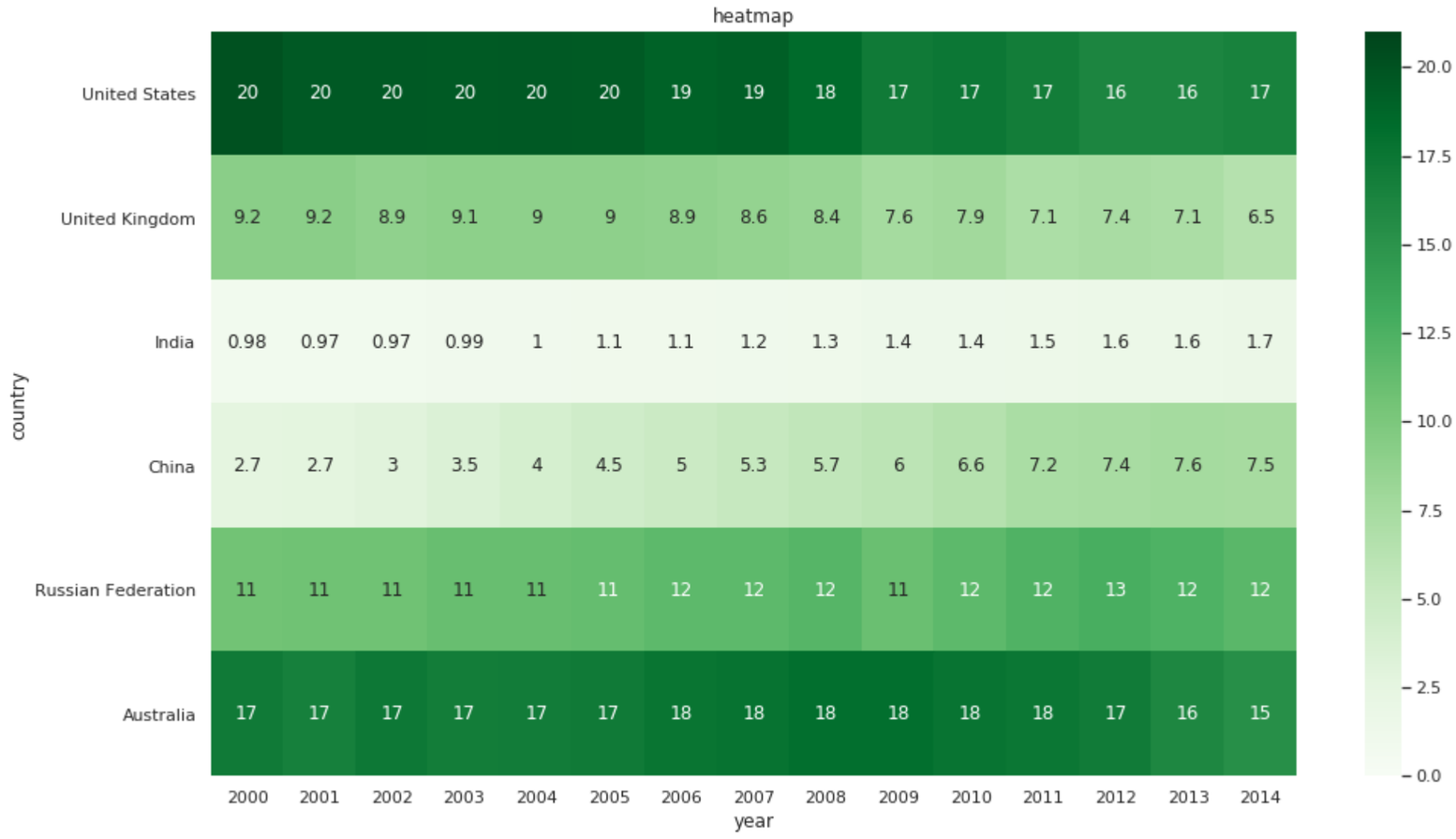


In [67]:
```
plt.figure(figsize=(16,9))
cbar_kws={

    "orientation":"horizontal",
    "shrink":1,
    "extend":"min",
    "extendfrac":0.1,
    "ticks":np.arange(0,22),
    "drawedges":True
}
sns.heatmap(hf,vmin=0,vmax=21,cmap="hot",annot=True,cbar_kws=cbar_kws)
```

Out[67]: <matplotlib.axes._subplots.AxesSubplot at 0x7f7b1dd4a310>



In [68]:
```
plt.figure(figsize=(16,9))
ax=sns.heatmap(hf,vmin=0,vmax=21,cmap="Greens",annot=True)
ax.set(title="heatmap",
       xlabel="year",
       ylabel="country")
sns.set(font_scale=1)
```



In [69]:
```
df=pd.read_csv("Who_is_responsible_for_global_warming.csv")
df
```

Out[69]:

	Country Name	Country Code	Indicator Name	Indicator Code	2000	2001	2002	2003	2004	2005	2006	2007	2008	2009	2010	2011	2012	2013	2014
0	United States	USA	CO2 emissions (metric tons per capita)	EN.ATM.CO2E.PC	20.178751	19.636505	19.613404	19.564105	19.658371	19.591885	19.094067	19.217898	18.461764	17.157738	17.442862	16.976957	16.310471	16.323477	16.502837
1	United Kingdom	GBR	CO2 emissions (metric tons per capita)	EN.ATM.CO2E.PC	9.199549	9.233175	8.904123	9.053278	8.989140	8.982939	8.898710	8.617164	8.424424	7.574622	7.857836	7.079298	7.355898	7.145844	6.497440
2	India	IND	CO2 emissions (metric tons per capita)	EN.ATM.CO2E.PC	0.979870	0.971698	0.967381	0.992392	1.025028	1.068563	1.121982	1.193210	1.310098	1.431844	1.397009	1.476686	1.598099	1.591438	1.730000
3	China	CHN	CO2 emissions (metric tons per capita)	EN.ATM.CO2E.PC	2.696862	2.742121	3.007083	3.524074	4.037991	4.523178	4.980314	5.334910	5.701915	6.010102	6.560520	7.241515	7.424751	7.557211	7.543908
4	Russian Federation	RUS	CO2 emissions (metric tons per capita)	EN.ATM.CO2E.PC	10.627121	10.669603	10.715901	11.090647	11.120627	11.253529	11.669122	11.672457	12.014507	11.023856	11.694348	12.334881	12.784979	12.393556	11.857528
5	Australia	AUS	CO2 emissions (metric tons per capita)	EN.ATM.CO2E.PC	17.200610	16.733367	17.370452	16.901959	17.026515	17.169711	17.651398	17.865260	18.160876	18.200182	17.740845	17.538878	17.072905	16.095833	15.388766
6	France	FRA	CO2 emissions (metric tons per capita)	EN.ATM.CO2E.PC	5.946665	6.153061	6.068664	6.115998	6.120079	6.099599	5.906266	5.766385	5.690501	5.438357	5.428981	5.077911	5.075064	5.062174	4.573182
7	Germany	DEU	CO2 emissions (metric tons per capita)	EN.ATM.CO2E.PC	10.095640	10.366287	10.058673	9.969355	9.898682	9.666372	9.911476	9.488040	9.506321	8.818596	9.279634	9.124859	9.199300	9.390623	8.889370
8	Canada	CAN	CO2 emissions (metric tons per capita)	EN.ATM.CO2E.PC	17.367115	16.985030	16.559378	17.461199	17.258911	17.251083	16.696694	16.855883	16.875198	15.961560	15.723167	15.639760	14.890636	14.711972	15.117159
9	Brazil	BRA	CO2 emissions (metric tons per capita)	EN.ATM.CO2E.PC	1.871118	1.898354	1.844380	1.762482	1.828672	1.858088	1.839394	1.901372	2.008670	1.883812	2.132938	2.211587	2.343570	2.488417	2.594388
10	Argentina	ARG	CO2 emissions (metric tons per capita)	EN.ATM.CO2E.PC	3.835574	3.568600	3.291548	3.525584	4.069058	4.141237	4.434821	4.382669	4.682912	4.410890	4.558500	4.600291	4.569384	4.462904	4.746797
11	Pakistan	PAK	CO2 emissions (metric tons per capita)	EN.ATM.CO2E.PC	0.768458	0.764702	0.788668	0.804959	0.872802	0.887768	0.929857	0.991030	0.972050	0.950832	0.946268	0.929801	0.918978	0.904316	0.896264

	Country Name	Country Code	Indicator Name	Indicator Code	2000	2001	2002	2003	2004	2005	2006	2007	2008	2009	2010	2011	2012	2013	2014
12	Nepal	NPL	CO2 emissions (metric tons per capita)	EN.ATM.CO2E.PC	0.129282	0.135226	0.106877	0.113902	0.105477	0.120277	0.098812	0.099736	0.129224	0.162087	0.187128	0.202491	0.211798	0.237170	0.283539
13	Bangladesh	BGD	CO2 emissions (metric tons per capita)	EN.ATM.CO2E.PC	0.211802	0.242020	0.246756	0.256602	0.266823	0.275247	0.299529	0.301631	0.332728	0.357159	0.393937	0.412011	0.433488	0.442401	0.459142
14	Japan	JPN	CO2 emissions (metric tons per capita)	EN.ATM.CO2E.PC	9.622352	9.464309	9.573130	9.725282	9.909203	9.698883	9.632049	9.782964	9.449534	8.620816	9.148316	9.317427	9.638628	9.780815	9.538706

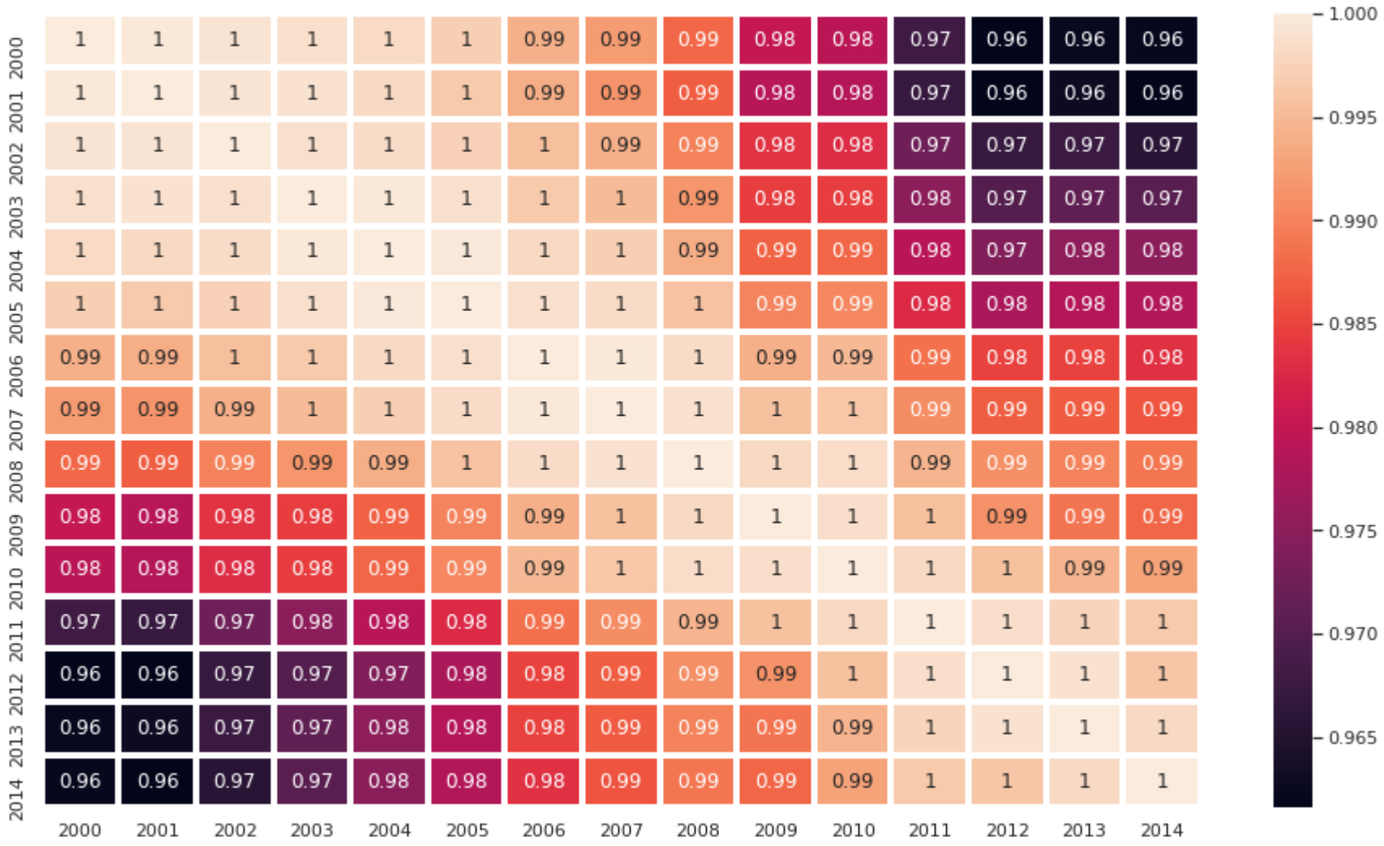
In [70]:
```
# correlation

df.corr()
```

Out[70]:

	2000	2001	2002	2003	2004	2005	2006	2007	2008	2009	2010	2011	2012	2013	2014
2000	1.000000	0.999632	0.999155	0.998911	0.998314	0.997008	0.994087	0.992283	0.987767	0.980143	0.979172	0.967887	0.961582	0.962466	0.962331
2001	0.999632	1.000000	0.999229	0.999026	0.998095	0.996628	0.993860	0.991532	0.987057	0.978912	0.978562	0.967206	0.961625	0.962827	0.961622
2002	0.999155	0.999229	1.000000	0.998907	0.998399	0.997391	0.995643	0.994017	0.990034	0.983584	0.982944	0.972479	0.967161	0.967573	0.965665
2003	0.998911	0.999026	0.998907	1.000000	0.999568	0.998887	0.996614	0.995277	0.991681	0.984511	0.984466	0.975128	0.969919	0.971053	0.970508
2004	0.998314	0.998095	0.998399	0.999568	1.000000	0.999701	0.998105	0.997144	0.993891	0.987300	0.987668	0.979061	0.974094	0.975276	0.975061
2005	0.997008	0.996628	0.997391	0.998887	0.999701	1.000000	0.998942	0.998420	0.995803	0.990125	0.990498	0.982646	0.977758	0.978611	0.978521
2006	0.994087	0.993860	0.995643	0.996614	0.998105	0.998942	1.000000	0.999570	0.998415	0.994104	0.994985	0.988553	0.984892	0.984857	0.983371
2007	0.992283	0.991532	0.994017	0.995277	0.997144	0.998420	0.999570	1.000000	0.999088	0.995724	0.996367	0.990928	0.986978	0.986819	0.986199
2008	0.987767	0.987057	0.990034	0.991681	0.993891	0.995803	0.998415	0.999088	1.000000	0.998145	0.998539	0.994593	0.991128	0.989983	0.988927
2009	0.980143	0.978912	0.983584	0.984511	0.987300	0.990125	0.994104	0.995724	0.998145	1.000000	0.998722	0.995657	0.991382	0.988844	0.987571
2010	0.979172	0.978562	0.982944	0.984466	0.987668	0.990498	0.994985	0.996367	0.998539	0.998722	1.000000	0.998182	0.995782	0.994553	0.992817
2011	0.967887	0.967206	0.972479	0.975128	0.979061	0.982646	0.988553	0.990928	0.994593	0.995657	0.998182	1.000000	0.998778	0.997744	0.996681
2012	0.961582	0.961625	0.967161	0.969919	0.974094	0.977758	0.984892	0.986978	0.991128	0.991382	0.995782	0.998778	1.000000	0.999066	0.996131
2013	0.962466	0.962827	0.967573	0.971053	0.975276	0.978611	0.984857	0.986819	0.989983	0.988844	0.994553	0.997744	0.999066	1.000000	0.998096
2014	0.962331	0.961622	0.965665	0.970508	0.975061	0.978521	0.983371	0.986199	0.988927	0.987571	0.992817	0.996681	0.996131	0.998096	1.000000

In [71]:
```
plt.figure(figsize=(16,9))
sns.heatmap(df.corr(),annot=True,linewidth=3)
```

Out[71]: <matplotlib.axes._subplots.AxesSubplot at 0x7f7b1dab4c90>

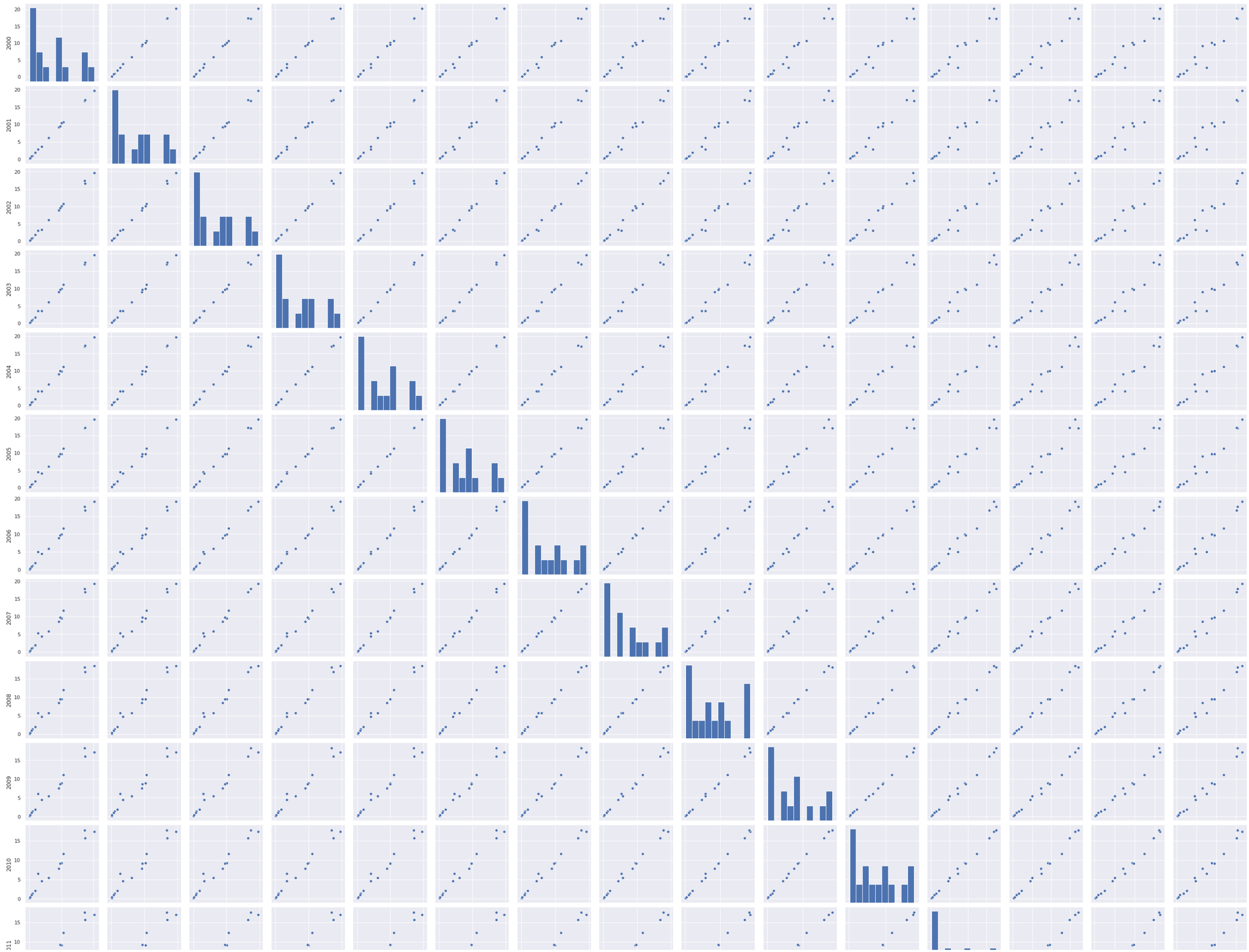


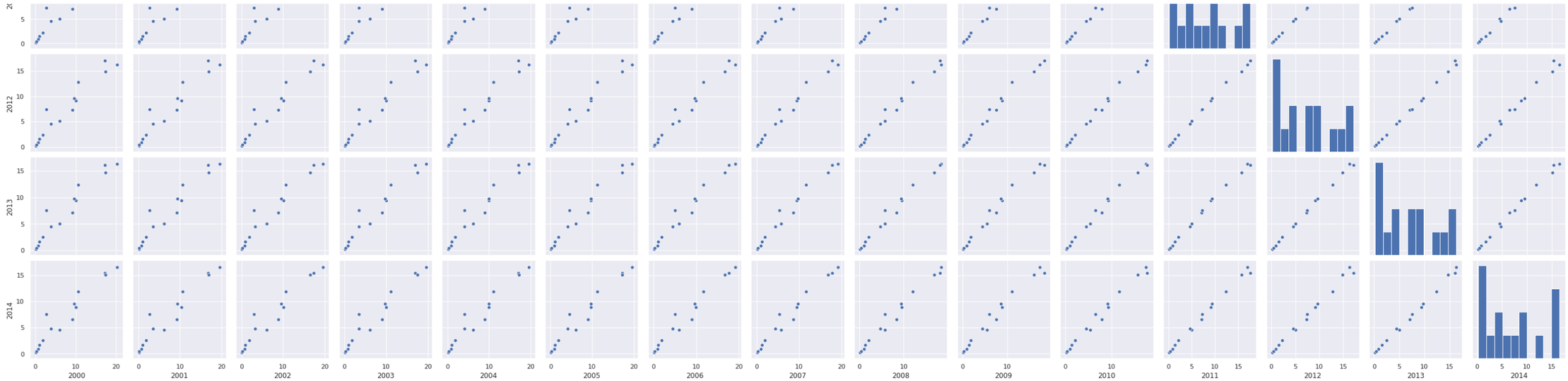
In [72]:
```
plt.figure(figsize=(16,9))
ax=sns.heatmap(df.corr(),annot=True,linewidth=3,cmap="hot")
ax.tick_params(size=10,color="green",labelsize=10,labelcolor="green")
```

pairplot

In [73]:
```
sns.pairplot(df)
```

Out[73]: <seaborn.axisgrid.PairGrid at 0x7f7b1d63ad90>


2000
2001
2002
2003
2004
2005
2006
2007
2008
2009
2010
2011
0
5
10
15
20



In [74]:
```
# set a var
pr=pd.read_csv("Text.csv")
pr
```

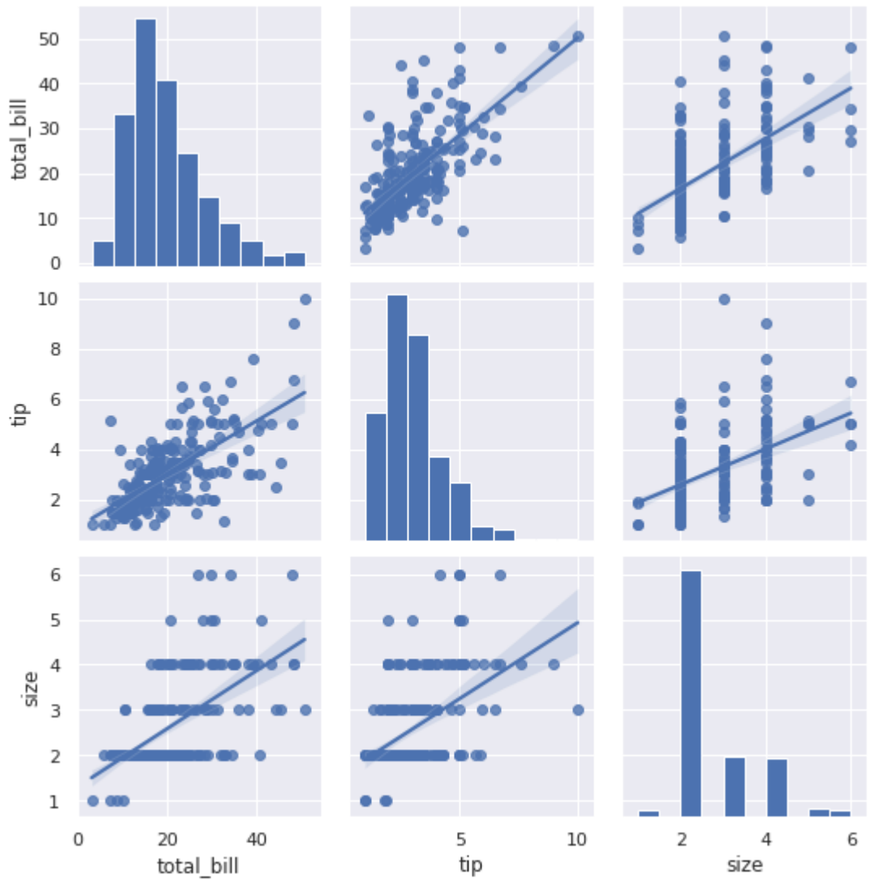
Out[74]:

	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 × 7 columns

In [75]:
```
sns.pairplot(pr,kind="reg")
```

Out[75]: <seaborn.axisgrid.PairGrid at 0x7f7b17ebcfd0>



In [ ]: