

seborn heatmap

```
import seaborn as sns # for data visualization
import pandas as pd # for data analysis
import numpy as np # for numeric calculation
import matplotlib.pyplot as plt # for data visualization
#import matplotlib

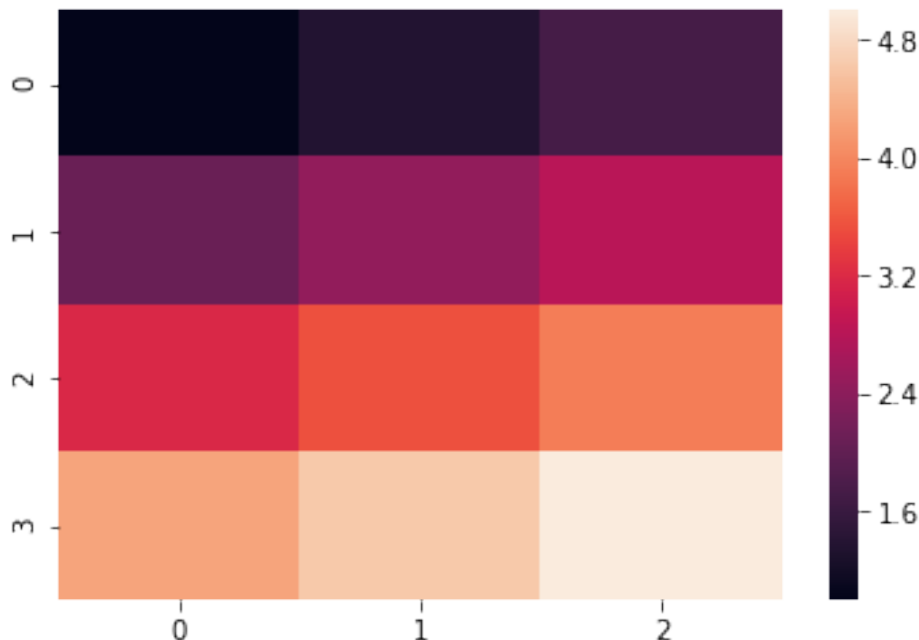
array_2d = np.linspace(1,5,12).reshape(4,3) # create numpy 2d array
print(array_2d) # print array

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

data

Seaborn Heatmap using Numpy 2D array

```
sns.heatmap(array_2d) # create histogram #img1 - *****
<matplotlib.axes._subplots.AxesSubplot at 0x2721170f128>
```



Seaborn Heatmap using Pandas DataFrame

```
# Load dataset ##### img2 -- ***** raw dataset
```

```
globalWarming_df =
```

```
pd.read_csv("Who_is_responsible_for_global_warming.csv")
globalWarming_df.head()
```

	Country Name	Country Code	Indicator
0	United States	USA	C02 emissions (metric tons per capita)
1	United Kingdom	GBR	C02 emissions (metric tons per capita)
2	India	IND	C02 emissions (metric tons per capita)
3	China	CHN	C02 emissions (metric tons per capita)
4	Russian Federation	RUS	C02 emissions (metric tons per capita)

	Indicator Code	2000	2001	2002	2003
0	EN.ATM.C02E.PC	20.178751	19.636505	19.613404	19.564105
1	EN.ATM.C02E.PC	9.199549	9.233175	8.904123	9.053278
2	EN.ATM.C02E.PC	0.979870	0.971698	0.967381	0.992392
3	EN.ATM.C02E.PC	2.696862	2.742121	3.007083	3.524074
4	EN.ATM.C02E.PC	10.627121	10.669603	10.715901	11.090647

	2005	2006	2007	2008	2009	2010
0	19.591885	19.094067	19.217898	18.461764	17.157738	17.442862
1	8.982939	8.898710	8.617164	8.424424	7.574622	7.857836
2	1.068563	1.121982	1.193210	1.310098	1.431844	1.397009
3	4.523178	4.980314	5.334910	5.701915	6.010102	6.560520
4	11.253529	11.669122	11.672457	12.014507	11.023856	11.694348

	2011	2012	2013	2014
0	16.976957	16.310471	16.323477	16.502837
1	7.079298	7.355898	7.145844	6.497440
2	1.476686	1.598099	1.591438	1.730000
3	7.241515	7.424751	7.557211	7.543908
4	12.334881	12.784979	12.393556	11.857528

```
##### img3 -- ***** dataset for heatmap
# set country name as index and drop Country Code      Indicator Name
Indicator Code
globalWarming_df = globalWarming_df.drop(columns=['Country Code',
'Indicator Name', 'Indicator Code'], axis=1).set_index('Country Name')
globalWarming_df
```

2004 \ Country Name	2000	2001	2002	2003
United States 19.658371	20.178751	19.636505	19.613404	19.564105
United Kingdom 8.989140	9.199549	9.233175	8.904123	9.053278
India 1.025028	0.979870	0.971698	0.967381	0.992392
China 4.037991	2.696862	2.742121	3.007083	3.524074
Russian Federation 11.120627	10.627121	10.669603	10.715901	11.090647
Australia 17.026515	17.200610	16.733367	17.370452	16.901959
France 6.120079	5.946665	6.153061	6.068664	6.115998
Germany 9.898682	10.095640	10.366287	10.058673	9.969355
Canada 17.258911	17.367115	16.985030	16.559378	17.461199
Brazil 1.828672	1.871118	1.898354	1.844380	1.762482
Argentina 4.069058	3.835574	3.568600	3.291548	3.525584
Pakistan 0.872802	0.768458	0.764702	0.788668	0.804959
Nepal 0.105477	0.129282	0.135226	0.106877	0.113902
Bangladesh 0.266823	0.211802	0.242020	0.246756	0.256602
Japan 9.909203	9.622352	9.464309	9.573130	9.725282

2009 \ Country Name	2005	2006	2007	2008
United States 17.157738	19.591885	19.094067	19.217898	18.461764
United Kingdom 7.574622	8.982939	8.898710	8.617164	8.424424
India 1.431844	1.068563	1.121982	1.193210	1.310098
China 6.010102	4.523178	4.980314	5.334910	5.701915
Russian Federation 11.023856	11.253529	11.669122	11.672457	12.014507
Australia	17.169711	17.651398	17.865260	18.160876

18.200182				
France	6.099599	5.906266	5.766385	5.690501
5.438357				
Germany	9.666372	9.911476	9.488040	9.506321
8.818596				
Canada	17.251083	16.696694	16.855883	16.875198
15.961560				
Brazil	1.858088	1.839394	1.901372	2.008670
1.883812				
Argentina	4.141237	4.434821	4.382669	4.682912
4.410890				
Pakistan	0.887768	0.929857	0.991030	0.972050
0.950832				
Nepal	0.120277	0.098812	0.099736	0.129224
0.162087				
Bangladesh	0.275247	0.299529	0.301631	0.332728
0.357159				
Japan	9.698883	9.632049	9.782964	9.449534
8.620816				

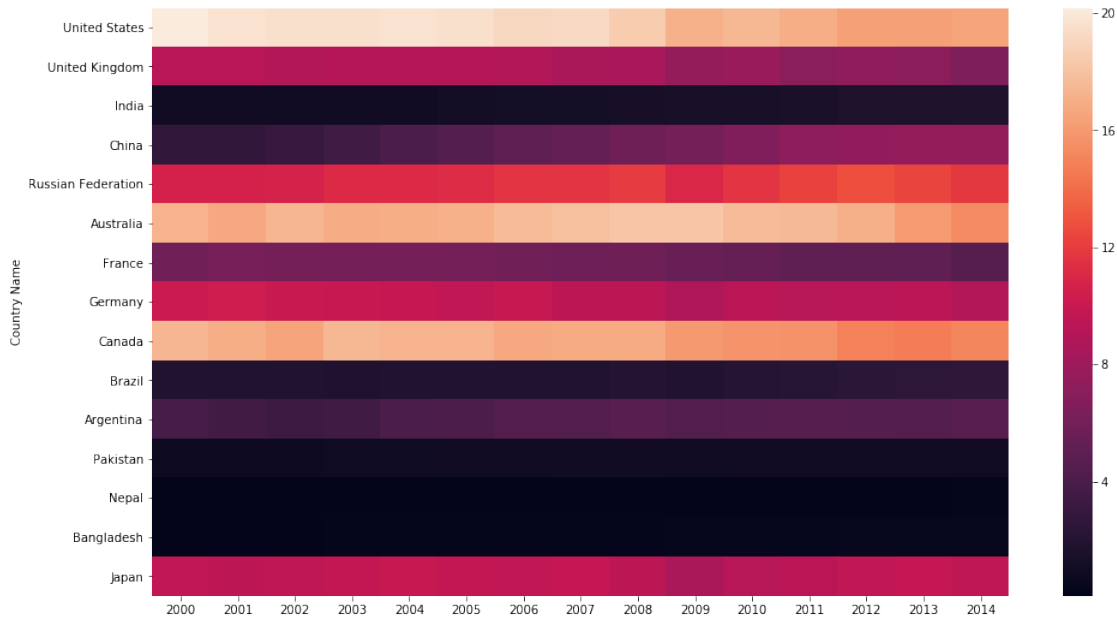
	2010	2011	2012	2013
2014				
Country Name				

United States	17.442862	16.976957	16.310471	16.323477
16.502837				
United Kingdom	7.857836	7.079298	7.355898	7.145844
6.497440				
India	1.397009	1.476686	1.598099	1.591438
1.730000				
China	6.560520	7.241515	7.424751	7.557211
7.543908				
Russian Federation	11.694348	12.334881	12.784979	12.393556
11.857528				
Australia	17.740845	17.538878	17.072905	16.095833
15.388766				
France	5.428981	5.077911	5.075064	5.062174
4.573182				
Germany	9.279634	9.124859	9.199300	9.390623
8.889370				
Canada	15.723167	15.639760	14.890636	14.711972
15.117159				
Brazil	2.132938	2.211587	2.343570	2.488417
2.594388				
Argentina	4.558500	4.600291	4.569384	4.462904
4.746797				
Pakistan	0.946268	0.929801	0.918978	0.904316
0.896264				
Nepal	0.187128	0.202491	0.211798	0.237170
0.283539				

Bangladesh	0.393937	0.412011	0.433488	0.442401
0.459142				
Japan	9.148316	9.317427	9.638628	9.780815
9.538706				

```
# Create heatmap ##### img4 -- ***** heatmap using DataFrame
plt.figure(figsize=(16,9))
sns.heatmap(globalWarming_df)
```

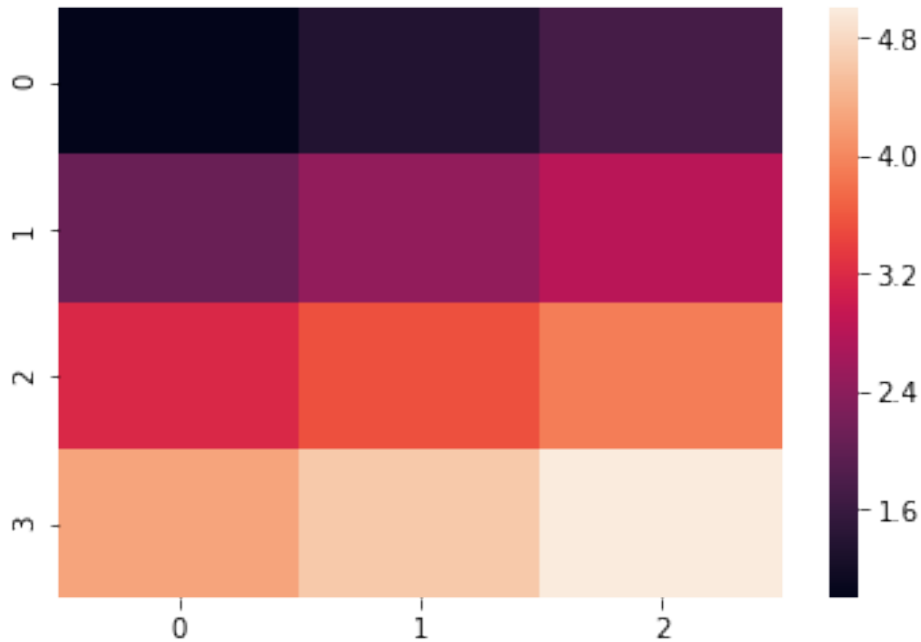
```
<matplotlib.axes._subplots.AxesSubplot at 0x27211a729b0>
```



vmin and vmax

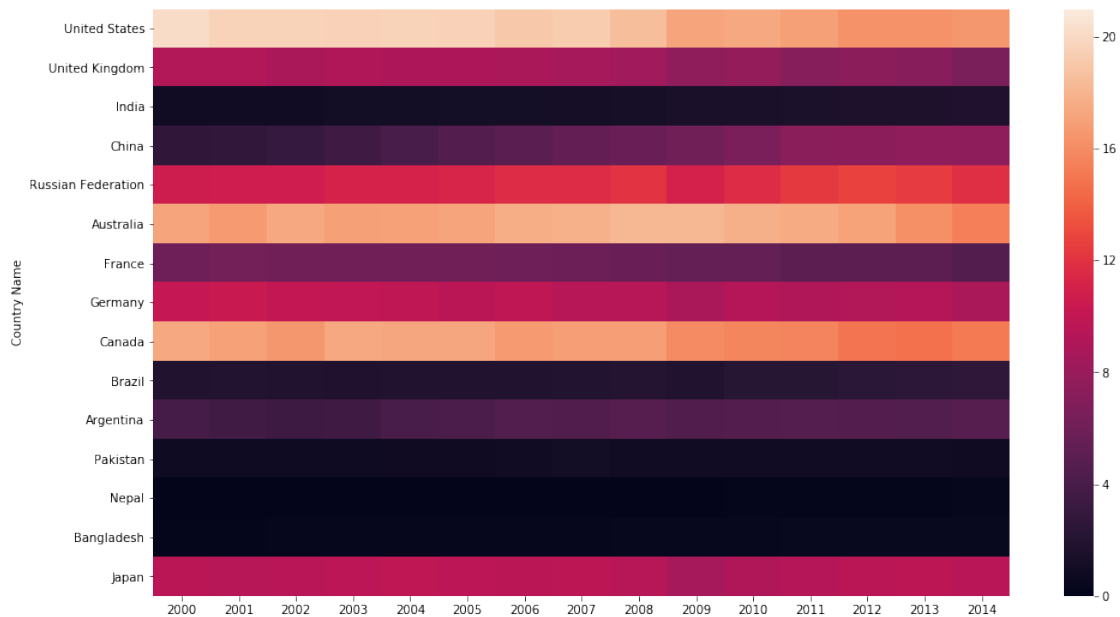
```
sns.heatmap(array_2d, vmin = 1, vmax = 5)
```

```
<matplotlib.axes._subplots.AxesSubplot at 0x27212e83f28>
```



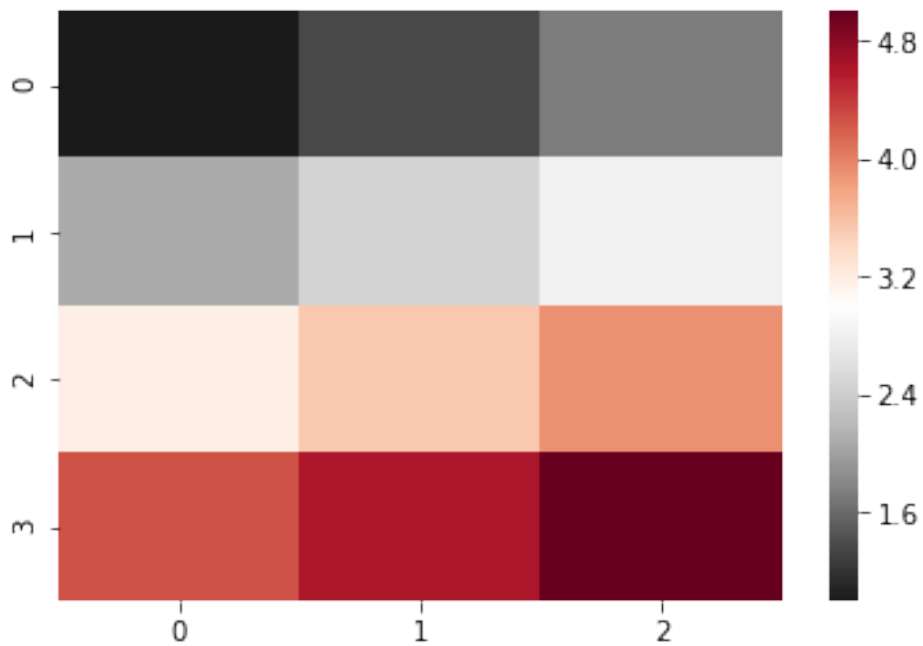
```
# Create heatmap ##### img5 -- ***** vmin & vmax
plt.figure(figsize=(16,9))
sns.heatmap(globalWarming_df, vmin = 0, vmax = 21)

<matplotlib.axes._subplots.AxesSubplot at 0x27212bcd9e8>
```



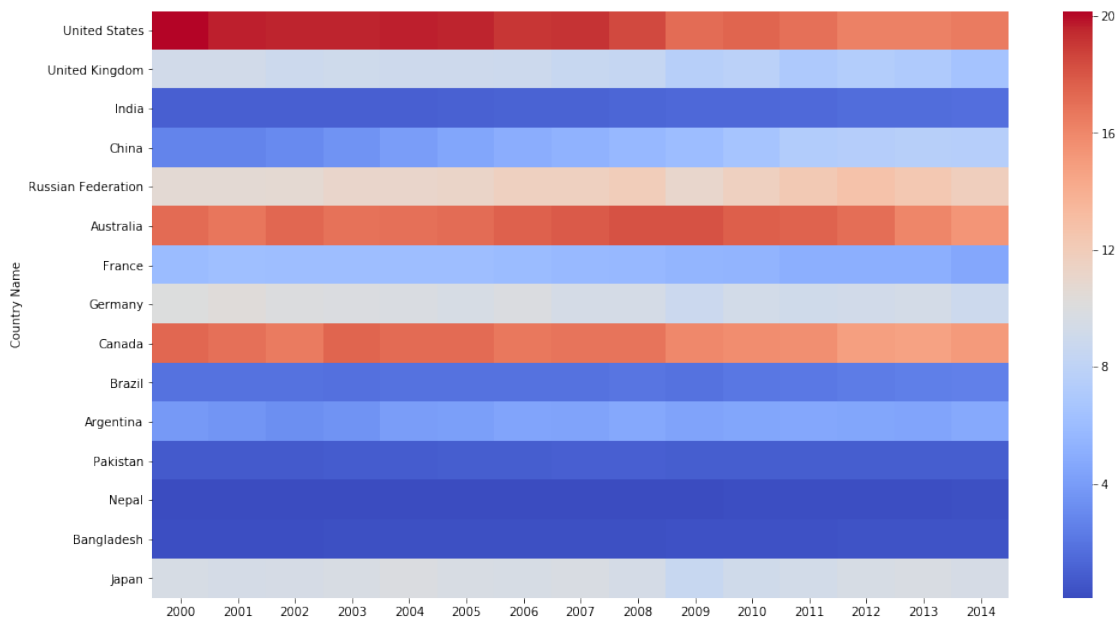
```
cmap
sns.heatmap(array_2d, vmin = 1, vmax = 5, cmap = "RdGy_r")
```

```
<matplotlib.axes._subplots.AxesSubplot at 0x27212ca2f28>
```



```
# change color map cmap ***** img 6 -- ***** cmap
plt.figure(figsize=(16,9))
sns.heatmap(globalWarming_df, cmap="coolwarm")
```

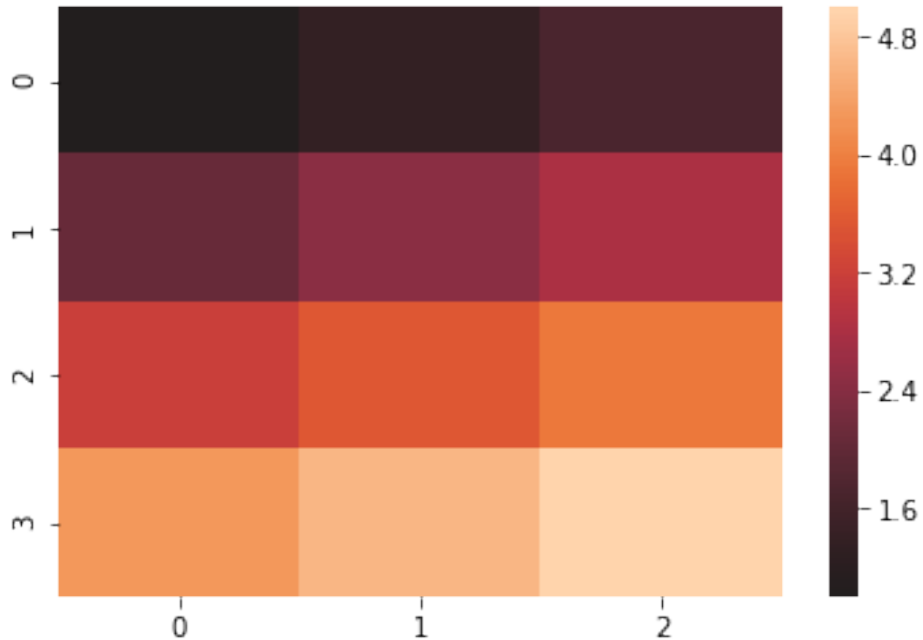
```
<matplotlib.axes._subplots.AxesSubplot at 0x27212d46668>
```



center

```
sns.heatmap(array_2d, vmin = 1, vmax = 5, center=0.90220)
```

```
<matplotlib.axes._subplots.AxesSubplot at 0x27212df8a90>
```

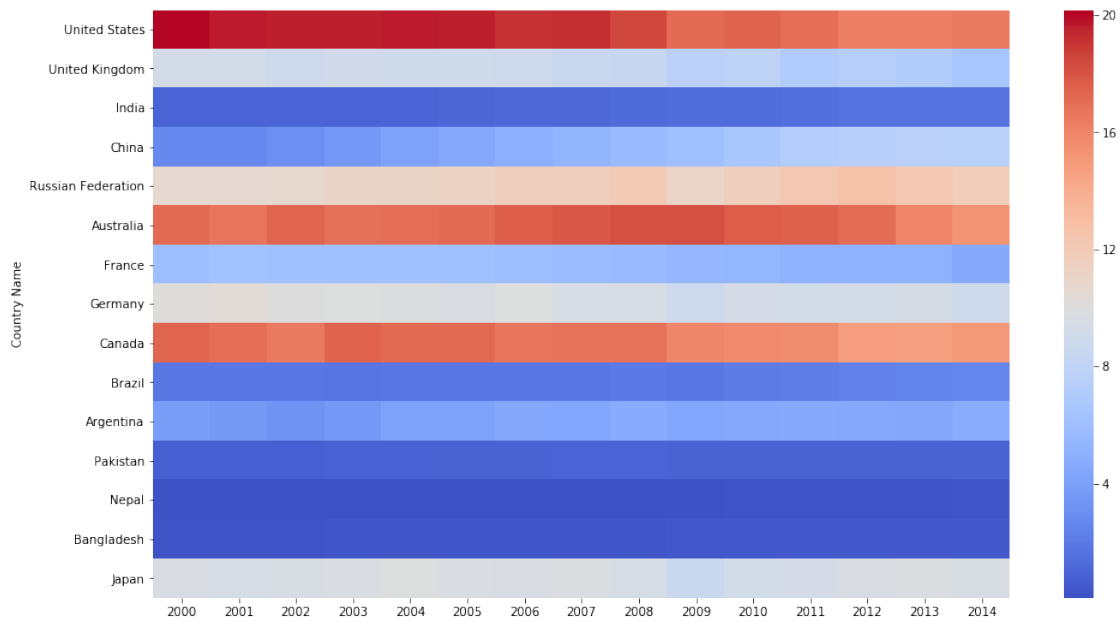


```
# center ### img 7 -- ***** center
```

```
plt.figure(figsize=(16,9))
```

```
sns.heatmap(globalWarming_df, cmap="coolwarm", center = 10.0)
```

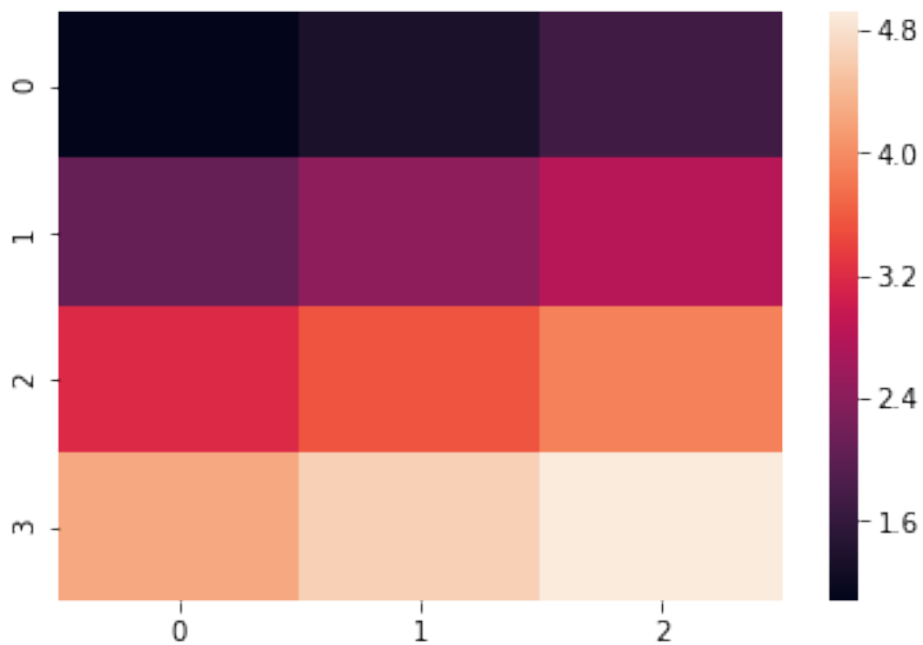
```
<matplotlib.axes._subplots.AxesSubplot at 0x272130f70f0>
```

robust

```
sns.heatmap(array_2d, robust= True)
```

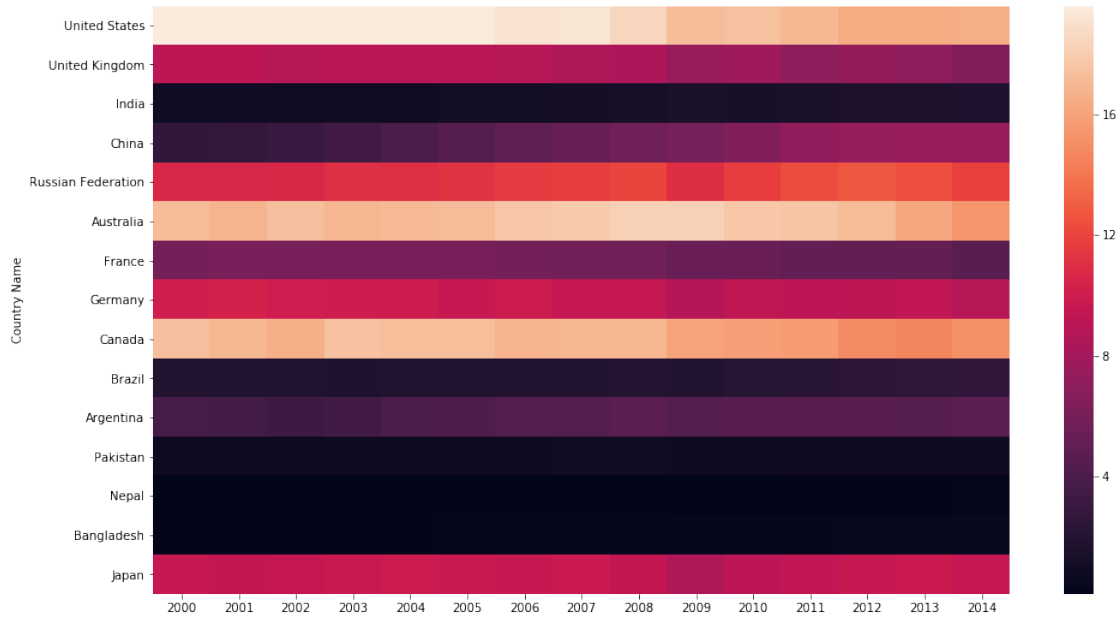
```
<matplotlib.axes._subplots.AxesSubplot at 0x272131bc3c8>
```



```
# robust ***** img 8 -- ***** robust
plt.figure(figsize=(16,9))
```

```
sns.heatmap(globalWarming_df, robust = True)
```

<matplotlib.axes._subplots.AxesSubplot at 0x2721323fb38>

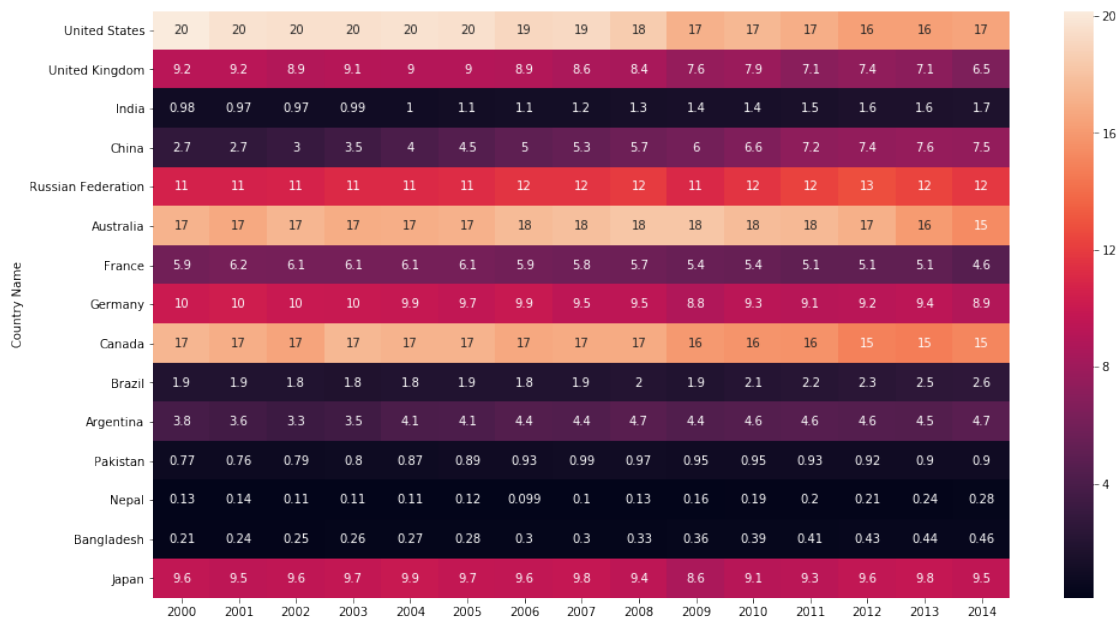


annot

```
# annot ***** img 9 -- ***** annot  
plt.figure(figsize=(16,9))
```

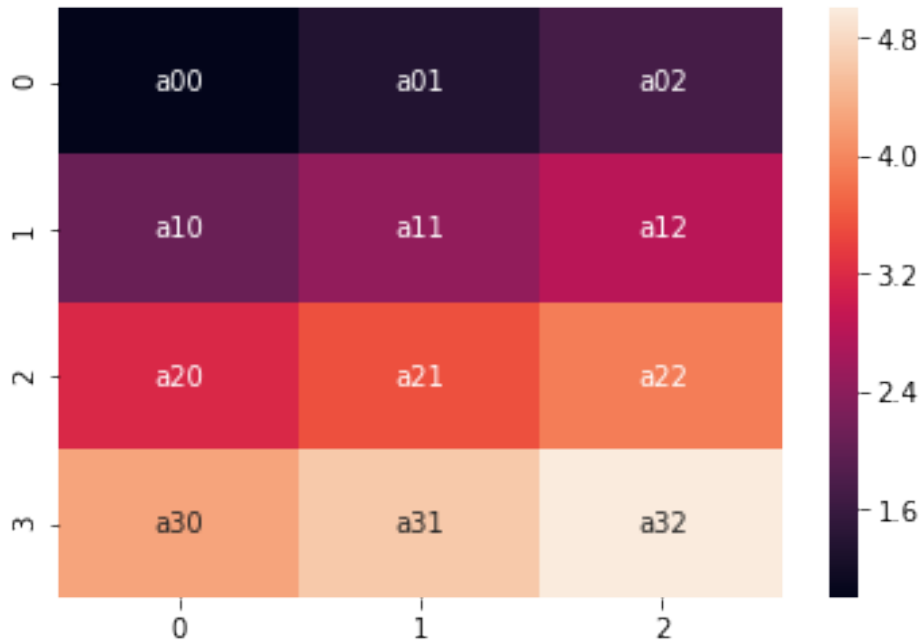
```
sns.heatmap(globalWarming_df, annot = True)
```

<matplotlib.axes._subplots.AxesSubplot at 0x272132f9ac8>




```
sns.heatmap(array_2d, annot = annot_arr, fmt="s") # s -string, d - decimal
```

```
<matplotlib.axes._subplots.AxesSubplot at 0x27213990a90>
```

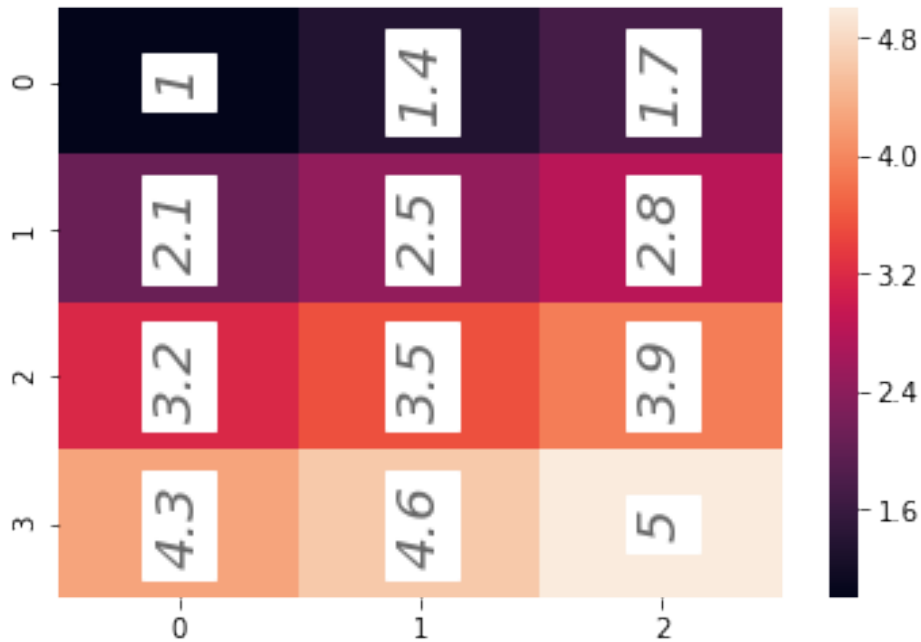


annot_kws

```
ax = sns.heatmap(array_2d, annot = True, annot_kws={'fontsize':20,  
'fontstyle':'italic', 'color':"k", 'alpha':0.6,  
'rotation':"vertical", 'verticalalignment':'center',  
                                                    'backgroundcolor':'w'})
```

#reference -

https://matplotlib.org/3.1.1/api/text_api.html#matplotlib.text.Text



```
# annot ***** img 12 -- ***** 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(globalWarming_df, annot = True, annot_kws= annot_kws)
<matplotlib.axes._subplots.AxesSubplot at 0x27213eb4e80>
```



linewidths

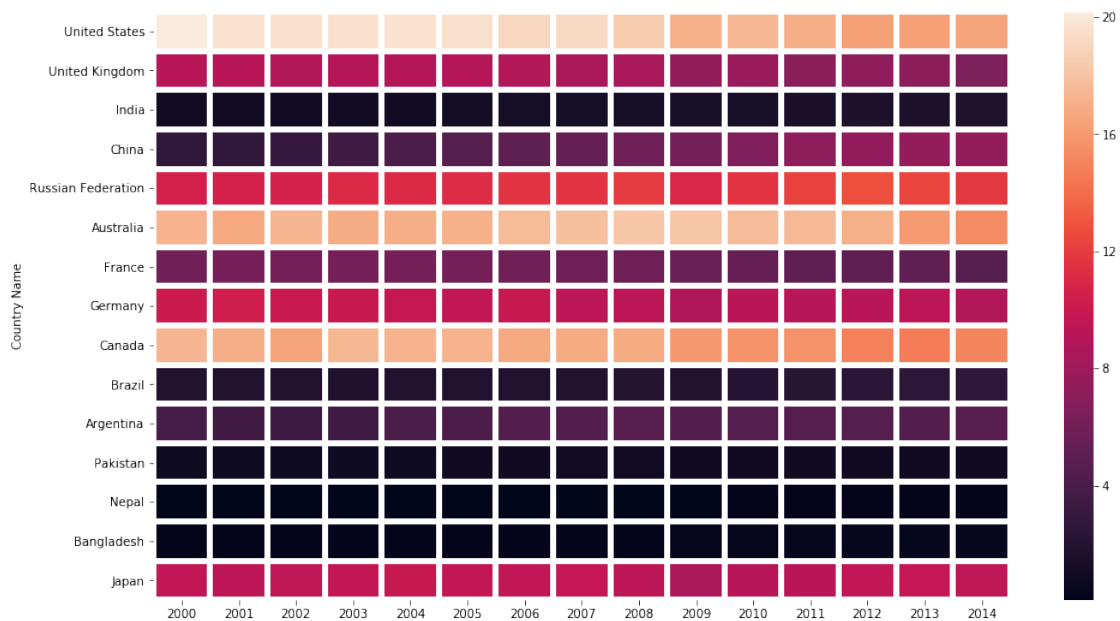
linewidths - divide each cell of heatmap ***** img 13 -- *

***** linewidths

```
plt.figure(figsize=(16,9))
```

```
sns.heatmap(globalWarming_df, linewidths=4)
```

<matplotlib.axes._subplots.AxesSubplot at 0x27213eb41d0>



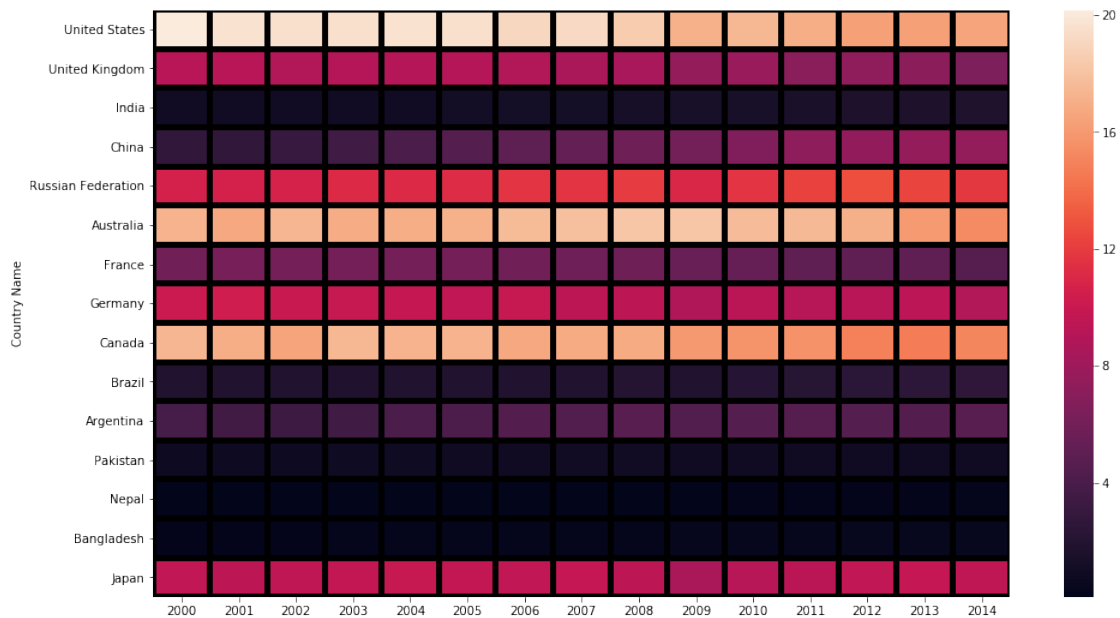
linecolor

```
# linecolor - change the color of heatmap line ***** img 14 -- *  
***** linecolor
```

```
plt.figure(figsize=(16,9))
```

```
sns.heatmap(globalWarming_df, linewidths=4, linecolor="k")
```

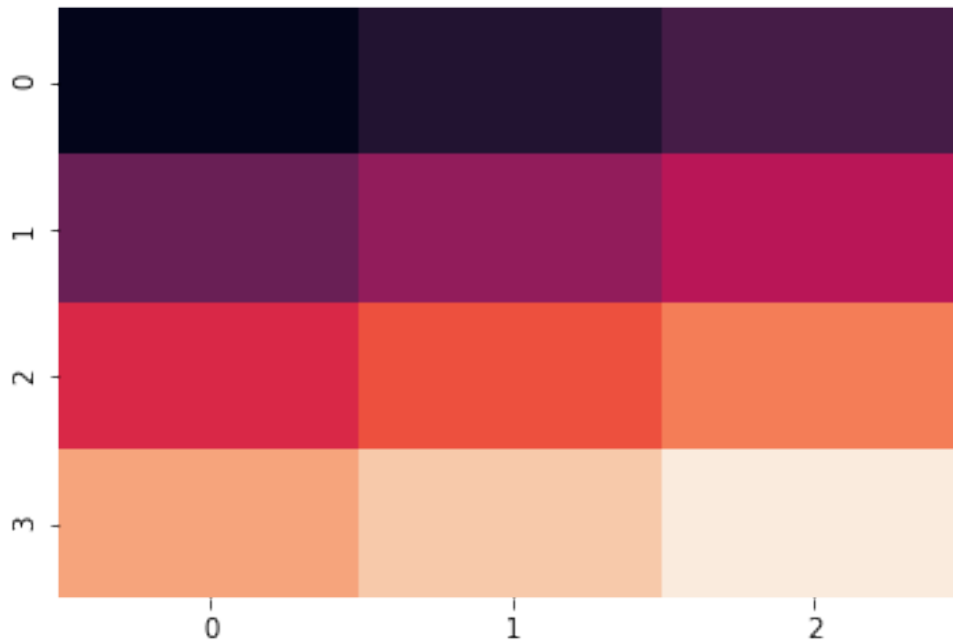
```
<matplotlib.axes._subplots.AxesSubplot at 0x27214314c50>
```



cbar

```
sns.heatmap(array_2d, cbar = False)
```

```
<matplotlib.axes._subplots.AxesSubplot at 0x272143368d0>
```

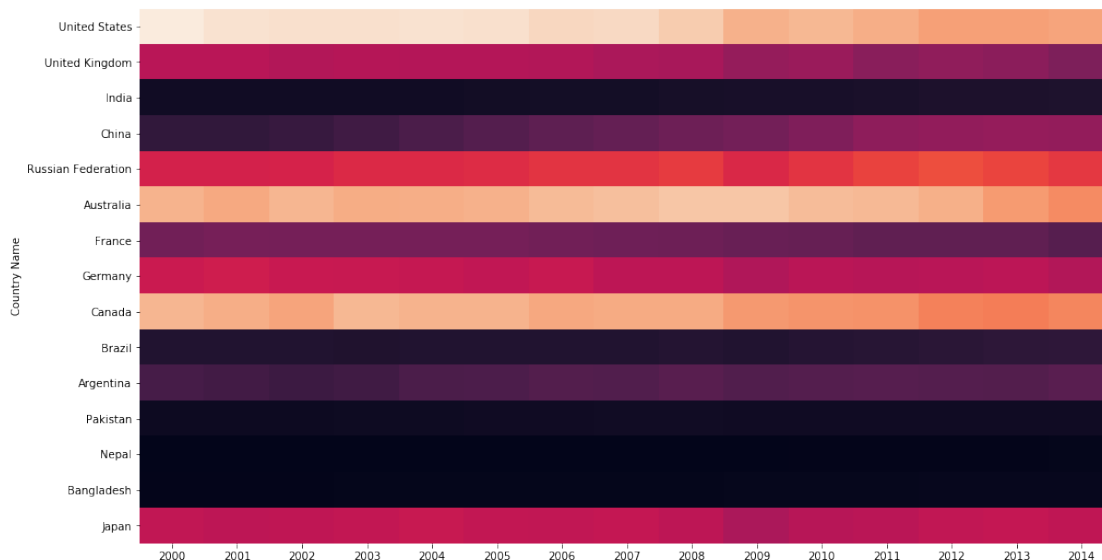


```
# hide color bar with cbar parameter ***** img 15 -- *
***** cbar
```

```
plt.figure(figsize=(16,9))
```

```
sns.heatmap(globalWarming_df, cbar = False)
```

```
<matplotlib.axes._subplots.AxesSubplot at 0x27214884898>
```



cbar_kws

```
#cbar_kws allow all properties of - plt.colorbar()
```

```
#https://matplotlib.org/3.1.1/api/_as_gen/matplotlib.pyplot.colorbar.h
```

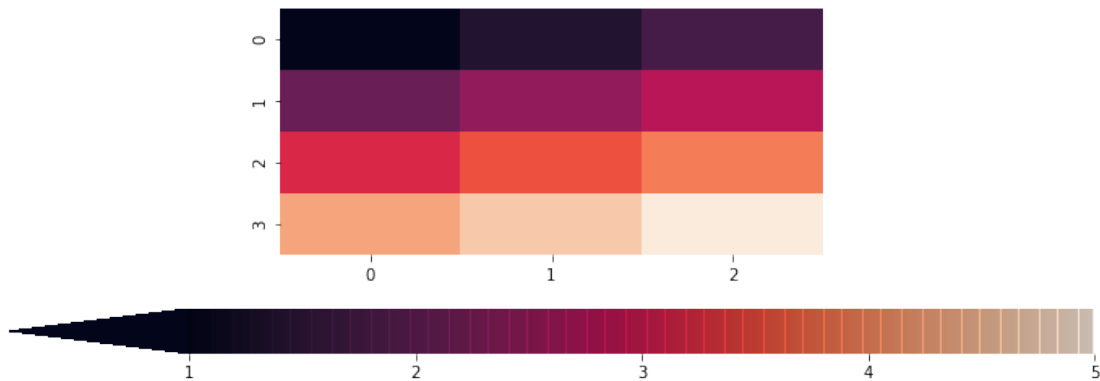


```
tml
```

```
cbar_kws = {"orientation": "horizontal", "shrink": 2, 'extend': 'min',  
'extendfrac': .2, "ticks": np.arange(1, 13),  
           "drawedges": True }
```

```
sns.heatmap(array_2d, cbar_kws=cbar_kws)  
# extend - [ 'neither' | 'both' | 'min' | 'max' ]
```

```
<matplotlib.axes._subplots.AxesSubplot at 0x2721490c390>
```



```
# change style and format of color bar with cbar_kws parameter
```

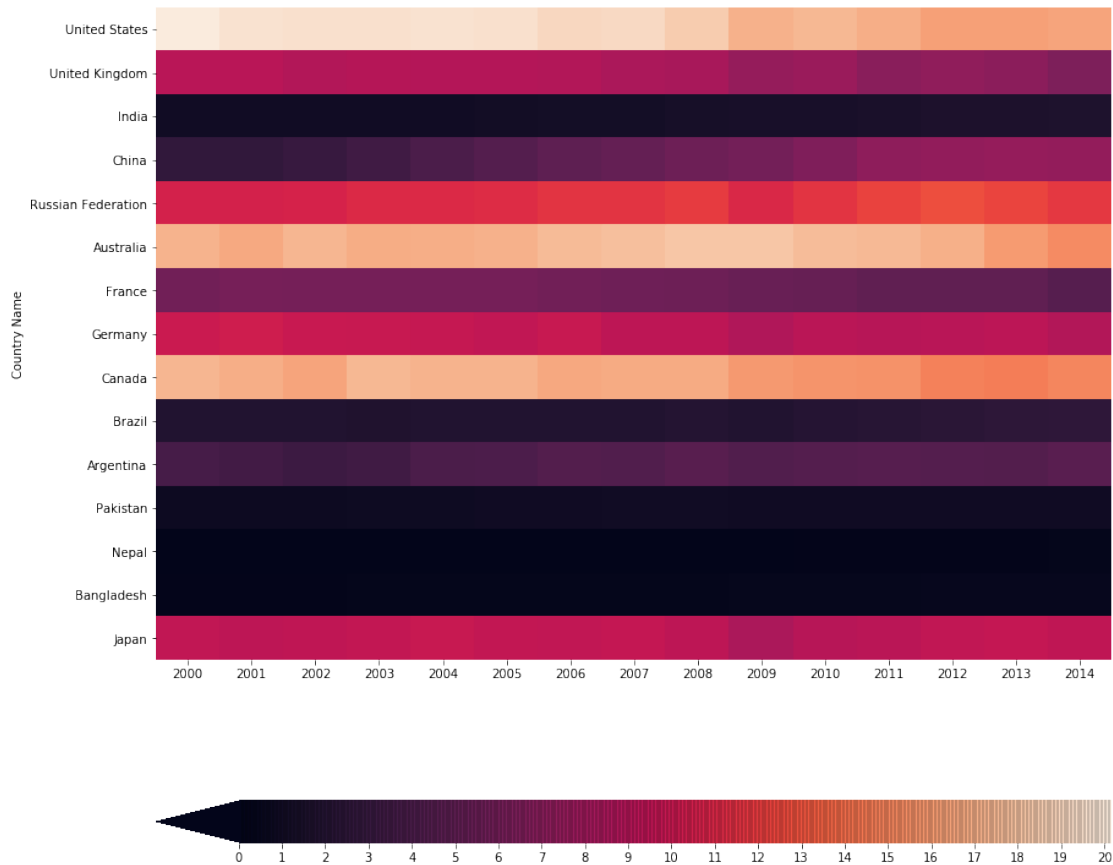
```
***** img 16 -- * ***** cbar_kws
```

```
plt.figure(figsize=(14, 14))
```

```
cbar_kws = {"orientation": "horizontal",  
           "shrink": 1,  
           'extend': 'min',  
           'extendfrac': 0.1,  
           "ticks": np.arange(0, 22),  
           "drawedges": True,  
           }
```

```
sns.heatmap(globalWarming_df, cbar_kws=cbar_kws)
```

```
<matplotlib.axes._subplots.AxesSubplot at 0x27214dec470>
```



cbar_ax

```
# need recerch
#ax = sns.heatmap(array_2d)
#ax.set(position='l')
```

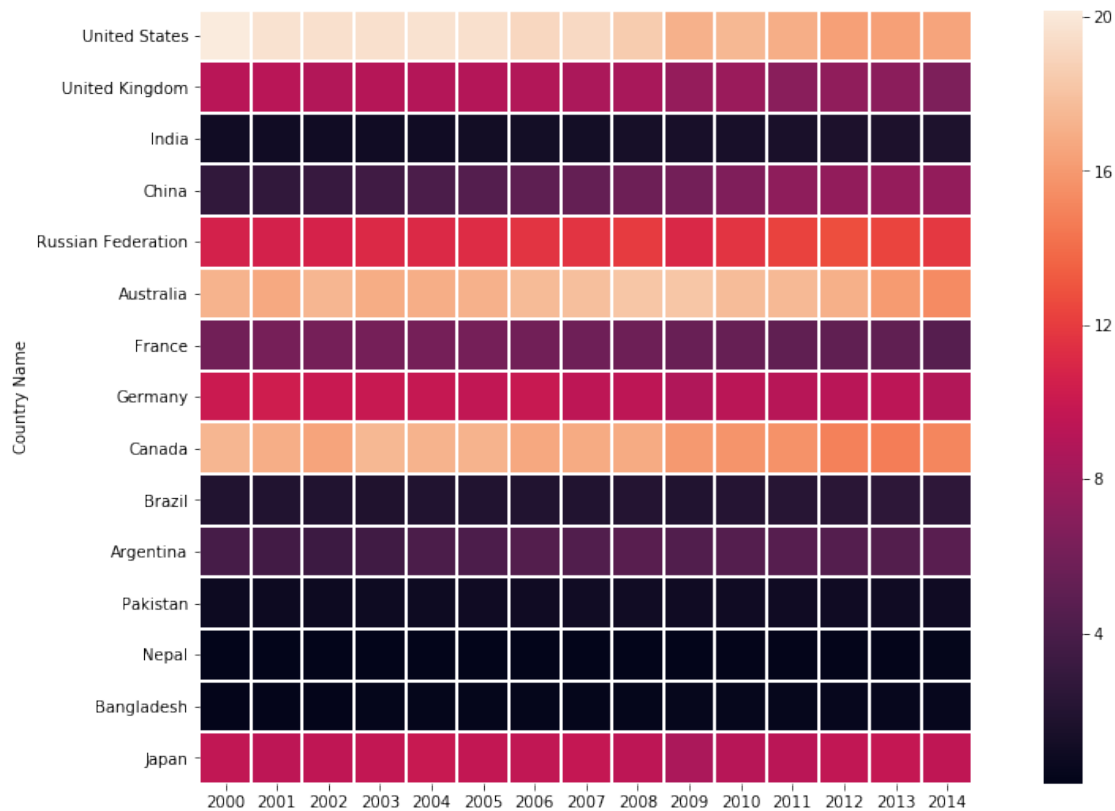
square

```
#sns.heatmap(int_n, square= True)

# Chage the shape of each cell ***** img 17 -- * *****
square
plt.figure(figsize=(16,9))

sns.heatmap(globalWarming_df, linewidths = 1, square= True)

<matplotlib.axes._subplots.AxesSubplot at 0x27214f123c8>
```



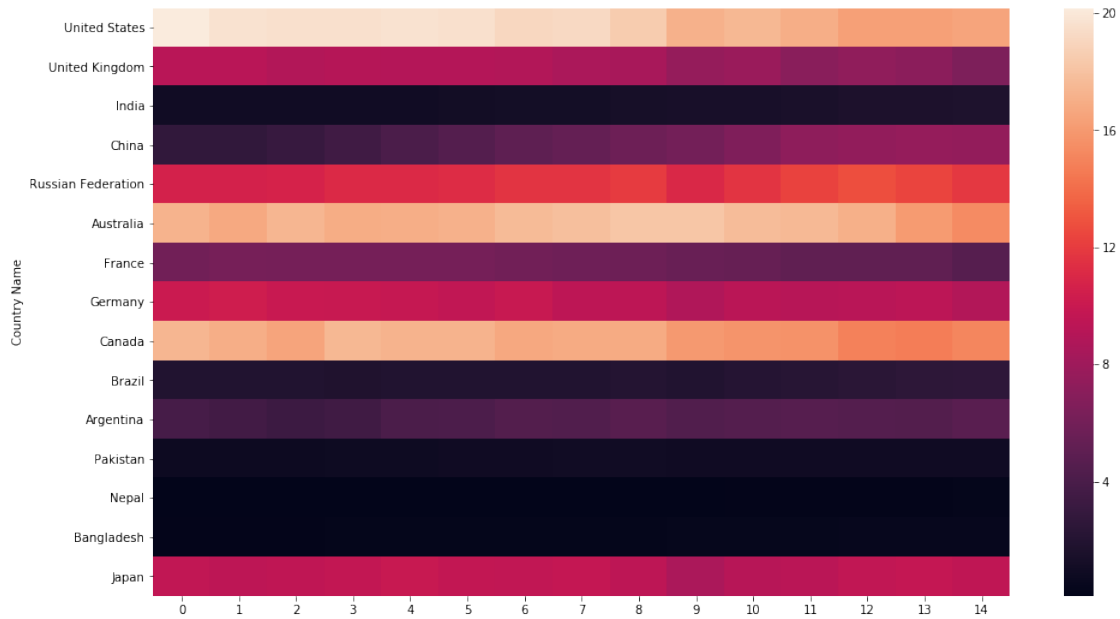
xticklabels

```
#sns.heatmap(int_n, xticklabels = ["colm-1","colm-2","colm-3"])

#sns.heatmap(int_n, xticklabels = False)

# change x-axis labels using xticklabels ***** img 18 -- *
***** xticklabels
plt.figure(figsize=(16,9))

sns.heatmap(globalWarming_df, xticklabels = np.arange(0,15))
<matplotlib.axes._subplots.AxesSubplot at 0x27214fd2240>
```



yticklabels

```
#sns.heatmap(int_n, yticklabels = ["row-1", "row-2", "row-3"])
```

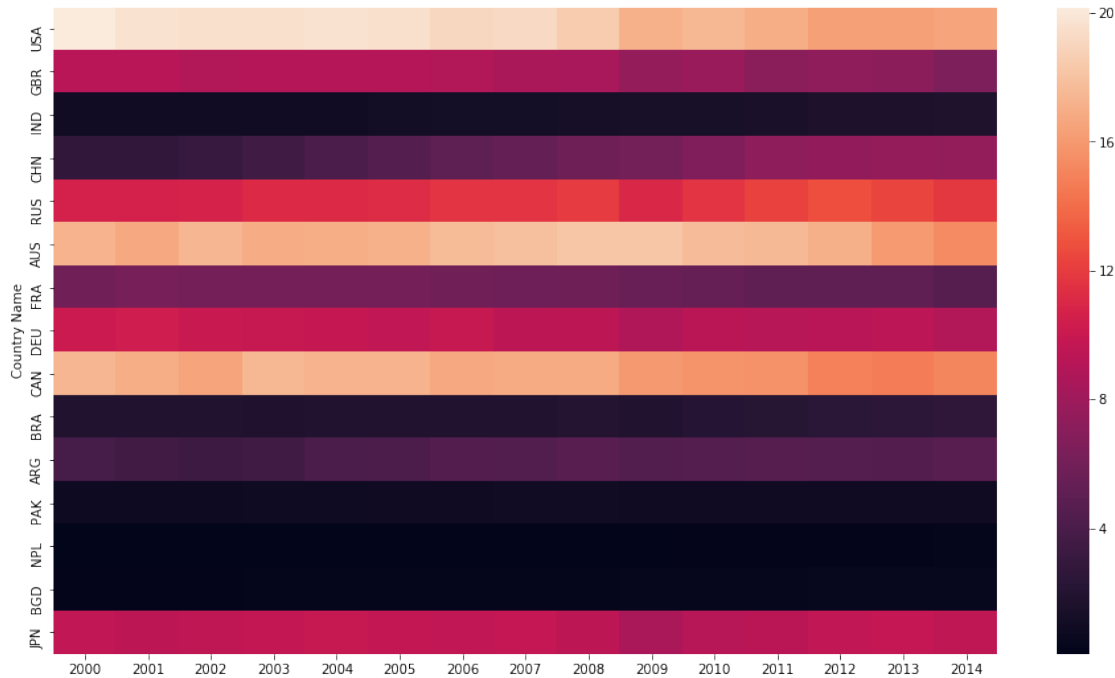
```
#sns.heatmap(int_n, yticklabels = False)
```

```
# change y-axis labels using yticklabels ***** img 19 -- *
***** yticklabels
```

```
plt.figure(figsize=(16,9))
```

```
country_code = ['USA', 'GBR', 'IND', 'CHN', 'RUS', 'AUS', 'FRA',
                'DEU', 'CAN', 'BRA', 'ARG', 'PAK', 'NPL', 'BGD', 'JPN']
sns.heatmap(globalWarming_df, yticklabels = country_code)
```

```
<matplotlib.axes._subplots.AxesSubplot at 0x27214b7c400>
```



Heatmap without xticklabels and yticklabels

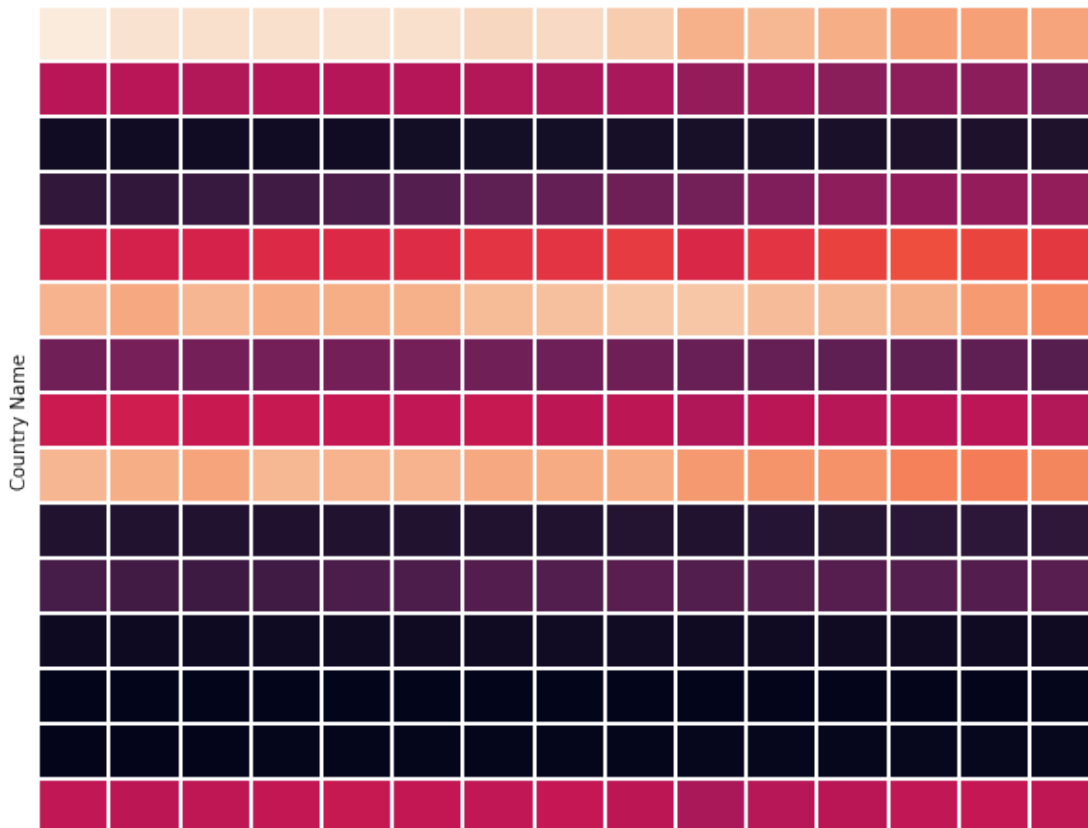
```
#sns.heatmap(globalWarming_df, xticklabels = False, yticklabels = False)
```

Heatmap without xticklabels, yticklabels and colorbar

```
# create square box ##### img 20 -- * only square
plt.figure(figsize=(10,8))
```

```
sns.heatmap(globalWarming_df, xticklabels = False, yticklabels = False, cbar=False, linecolor="w", linewidths=1)
```

```
<matplotlib.axes._subplots.AxesSubplot at 0x272160fbc88>
```



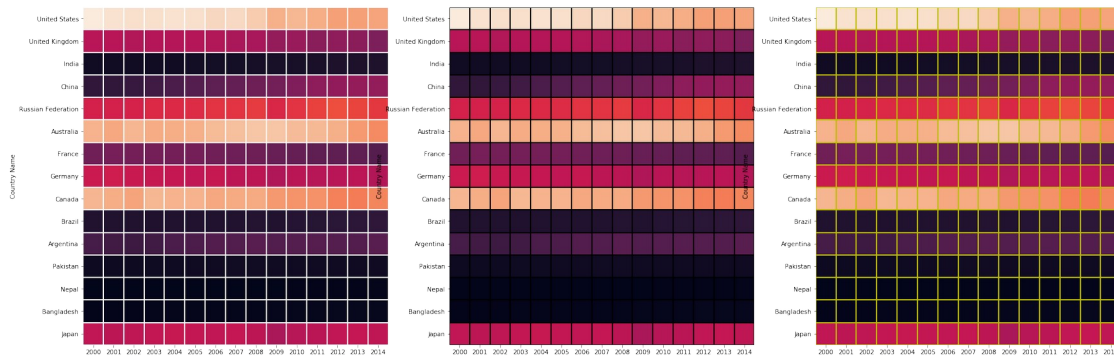
Subplots - create multiple heatmaps

*# multiple heatmaps using subplots ##### img 21 -- * subplots*
`plt.figure(figsize=(30,10))`

`plt.subplot(1,3,1) # first heatmap`
`sns.heatmap(globalWarming_df, cbar=False, linecolor="w",`
`linewidths=1)`

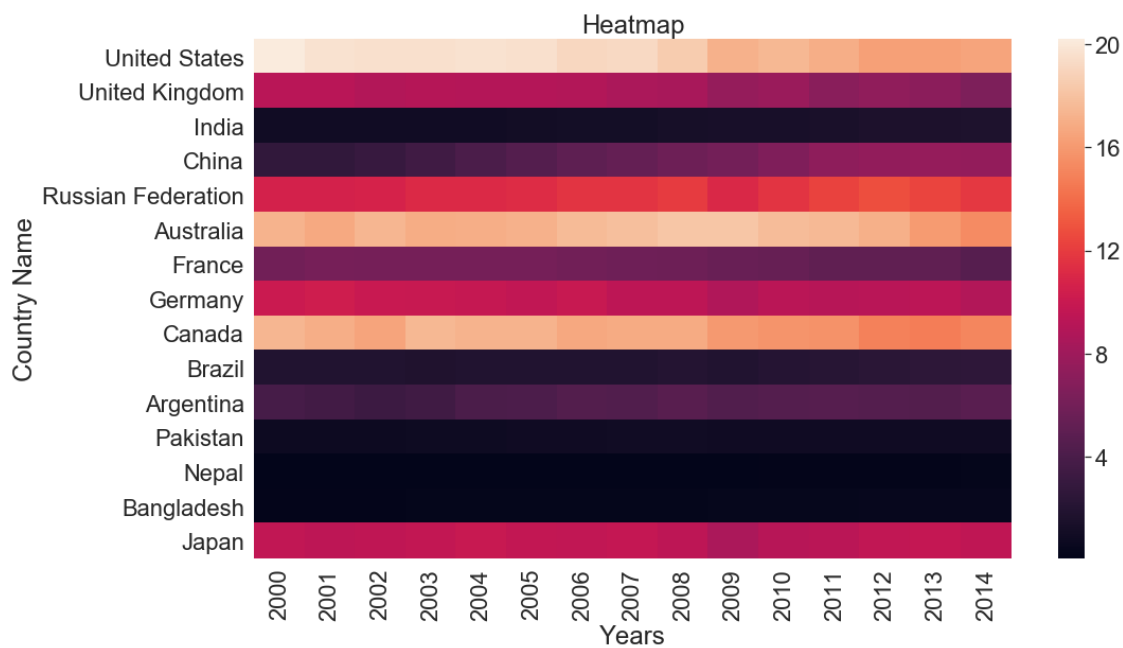
`plt.subplot(1,3,2) # second heatmap`
`sns.heatmap(globalWarming_df, cbar=False, linecolor="k",`
`linewidths=1)`

`plt.subplot(1,3,3) # third heatmap`
`sns.heatmap(globalWarming_df, cbar=False, linecolor="y",`
`linewidths=1)`
`plt.show()`



ax - Axes

```
# set seaborn heatmap title, x-axis, y-axis label and font size
##### img 22 -- * ax
plt.figure(figsize=(16,9))
ax = sns.heatmap(globalWarming_df,)
ax.set(title="Heatmap",
        xlabel="Years",
        ylabel="Country Name",)
sns.set(font_scale=2) # set fontsize 2
```



kwargs

```
# Heatmap with keyword arguments (kwargs) ##### img 23 -- *
kwargs
plt.figure(figsize = (16,9))

kwargs = {'alpha':.9, 'linewidth':10, 'linestyle':'--',
```

```
'rasterized':False, 'edgecolor':'w', "capstyle":'projecting',}
```

```
ax = sns.heatmap(globalWarming_df,**kwargs)
```



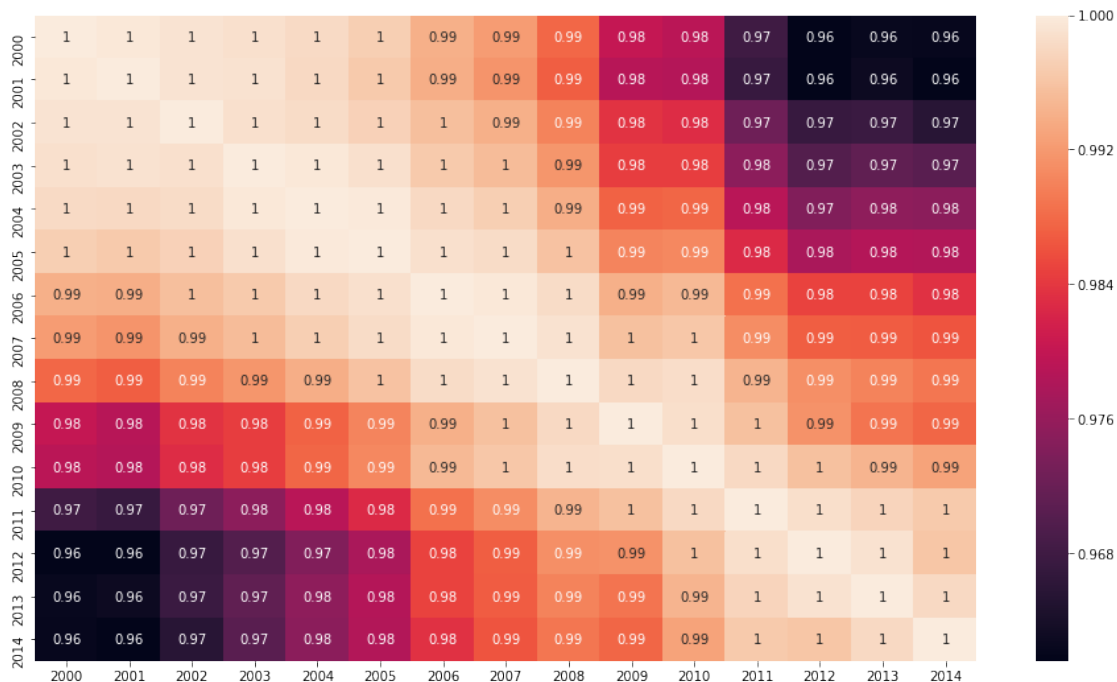
correlation

```
# sns heatmap correlation
```

```
plt.figure(figsize=(16,9))
```

```
sns.heatmap(globalWarming_df.corr(), annot = True)
```

```
<matplotlib.axes._subplots.AxesSubplot at 0x27216da09e8>
```

mask

*# Upper triangle heatmap ##### img 25 -- * upper heatmap*

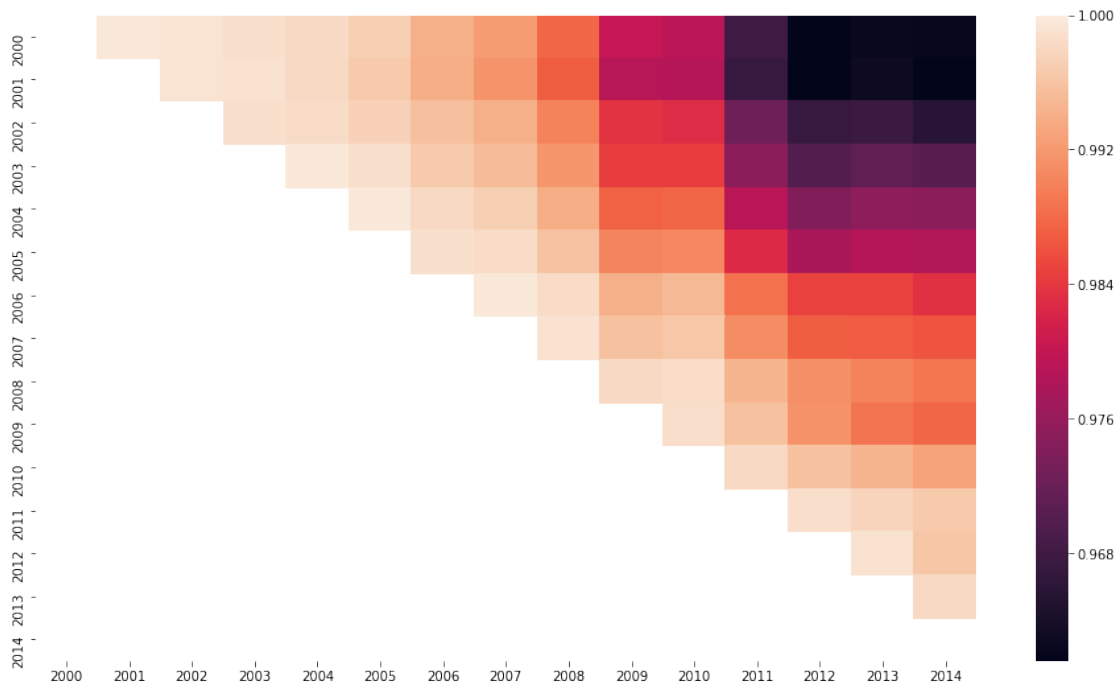
```
plt.figure(figsize=(16,9))
```

```
corr_mx = globalWarming_df.corr() # correlation matrix
```

```
matrix = np.tril(corr_mx) # take lower correlation matrix
```

```
sns.heatmap(corr_mx, mask=matrix)
```

```
<matplotlib.axes._subplots.AxesSubplot at 0x27217ac8940>
```



```
# Lower triangle heatmap ##### img 26 -- * lower heatmap
```

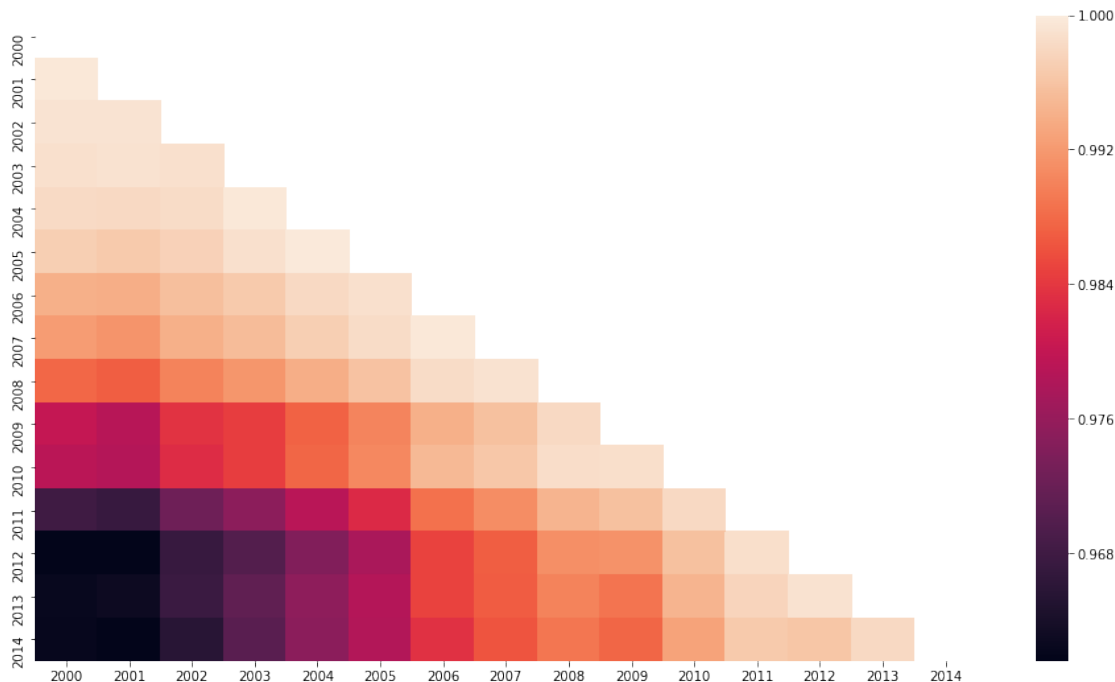
```
plt.figure(figsize=(16,9))
```

```
corr_mx = globalWarming_df.corr() # correlation matrix
```

```
matrix = np.triu(corr_mx) # take upper correlation matrix
```

```
sns.heatmap(corr_mx, mask=matrix)
```

```
<matplotlib.axes._subplots.AxesSubplot at 0x27216a84898>
```



Examples

import libraries

import seaborn **as** sns *# for data visualization*

import matplotlib.pyplot **as** plt *# for data visualization*

import pandas **as** pd *# for data analysis*

load dataset and create DataFrame ready to create heatmap

globalWarming_df =

pd.read_csv("Who_is_responsible_for_global_warming.csv")

globalWarming_df = globalWarming_df.drop(columns=['Country Code',
'Indicator Name', 'Indicator Code'], axis=1).set_index('Country Name')

set heatmap size

plt.figure(figsize= (16,9))

create heatmap seaborn

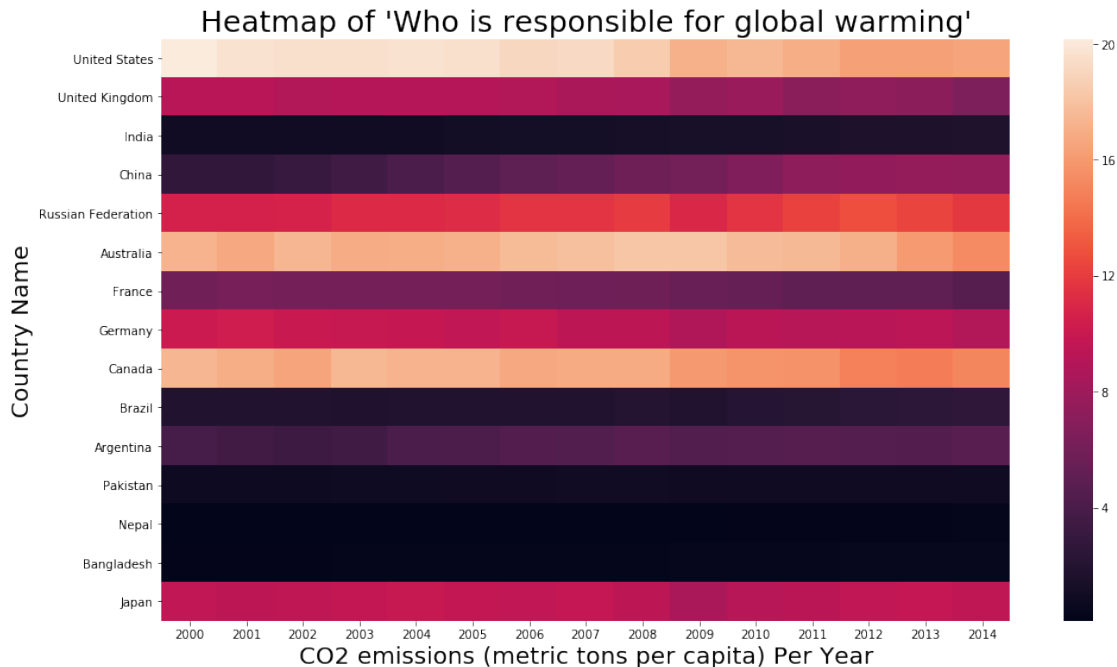
sns.heatmap(globalWarming_df)

plt.title("Heatmap of 'Who is responsible for global warming'",
fontsize = 25)

plt.xlabel("CO2 emissions (metric tons per capita) Per Year", fontsize
= 20)

plt.ylabel("Country Name", fontsize = 20)

plt.show()



Example 2:

```
# import libraries
import seaborn as sns # for data visualization
import matplotlib.pyplot as plt # for data visualization
import pandas as pd # for data analysis

# load dataset and create DataFrame ready to create heatmap
globalWarming_df =
pd.read_csv("Who_is_responsible_for_global_warming.csv")
globalWarming_df = globalWarming_df.drop(columns=['Country Code',
'Indicator Name', 'Indicator Code'], axis=1).set_index('Country Name')

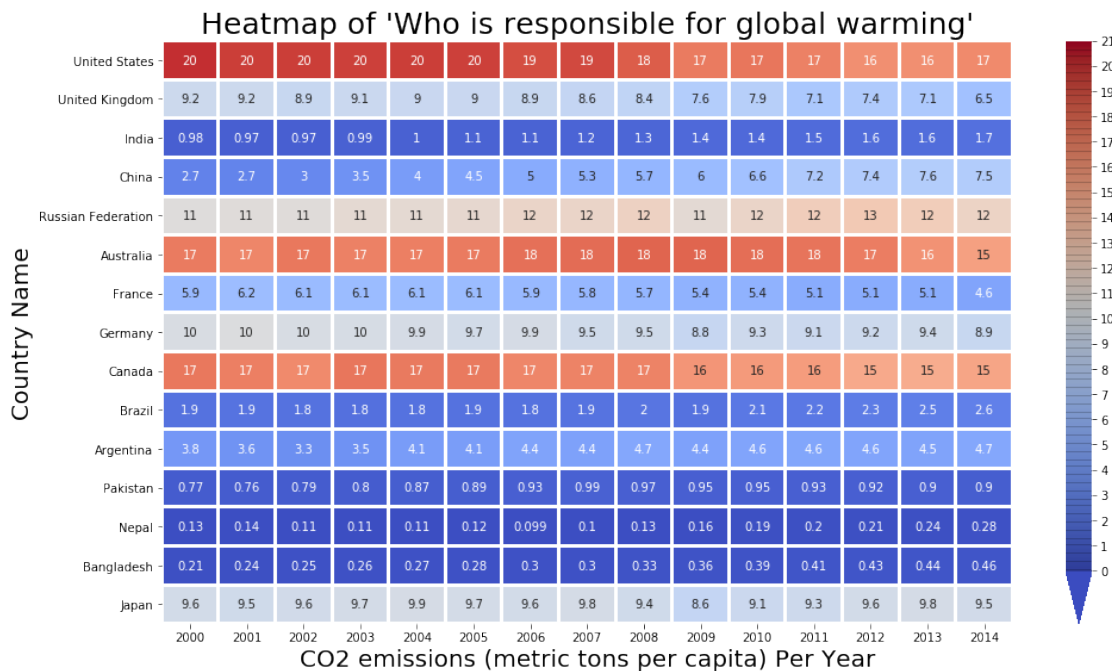
# set heatmap size
plt.figure(figsize= (16,9))

# create heatmap seaborn

cbar_kws = {"orientation":"vertical",
            "shrink":1,
            'extend':'min',
            'extendfrac':0.1,
            "ticks":np.arange(0,22),
            "drawedges":True,
            } # color bar keyword arguments

sns.heatmap(globalWarming_df, vmin = 0, vmax = 21, cmap="coolwarm",
annot = True, linewidth = 2, cbar_kws=cbar_kws)
```

```
plt.title("Heatmap of 'Who is responsible for global warming'",
         fontsize = 25)
plt.xlabel("CO2 emissions (metric tons per capita) Per Year", fontsize
         = 20)
plt.ylabel("Country Name", fontsize = 20)
plt.show()
```



Example 3

```
# import libraries
import seaborn as sns # for data visualization
import matplotlib.pyplot as plt # for data visualization
import pandas as pd # for data analysis

# load dataset and create DataFrame ready to create heatmap
flights = sns.load_dataset("flights")
flights_df = flights.pivot("month", "year", "passengers")

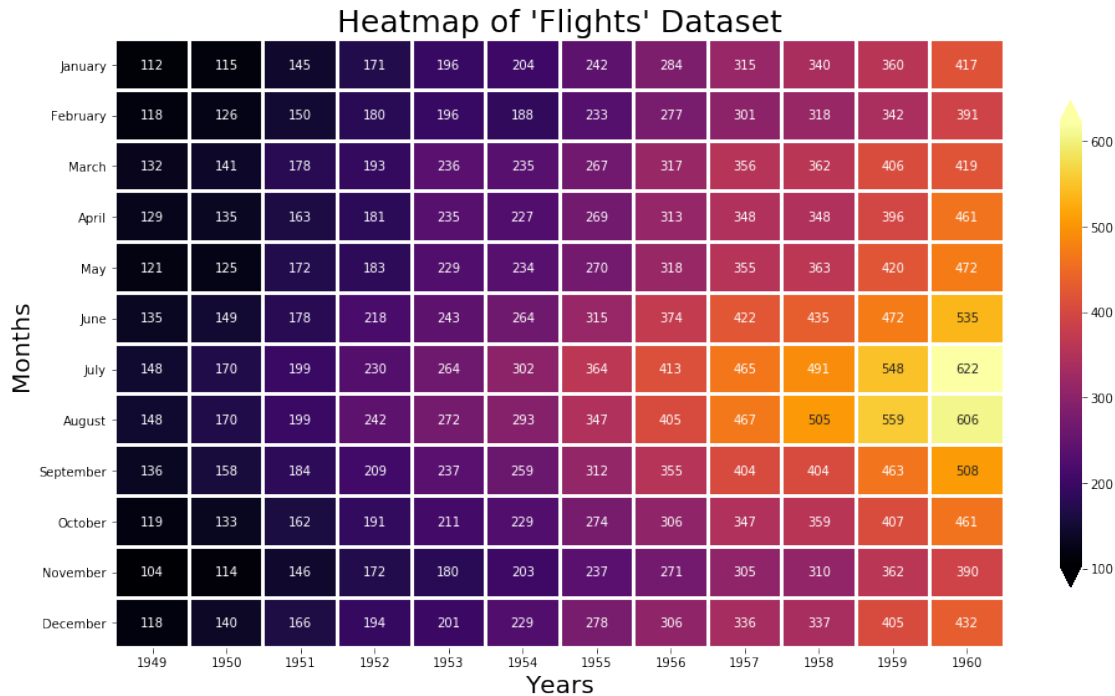
# set heatmap size
plt.figure(figsize= (16,9))

# create heatmap seaborn

cbar_kws = {"shrink":.8,
            'extend':'both'}

sns.heatmap(flights_df, cmap="inferno", annot = True, fmt = 'd',
            linewidth = 2, cbar_kws=cbar_kws)
```

```
plt.title("Heatmap of 'Flights' Dataset", fontsize = 25)
plt.xlabel("Years", fontsize = 20)
plt.ylabel("Months", fontsize = 20)
plt.show()
```



Examples 4 # correlation

```
# import libraries
```

```
import seaborn as sns # for data visualization
```

```
import matplotlib.pyplot as plt # for data visualization
```

```
import pandas as pd # for data analysis
```

```
# load dataset and create DataFrame ready to create heatmap
```

```
flights = sns.load_dataset("flights")
```

```
flights_df = flights.pivot("month", "year", "passengers")
```

```
# set heatmap size
```

```
plt.figure(figsize= (16,9))
```

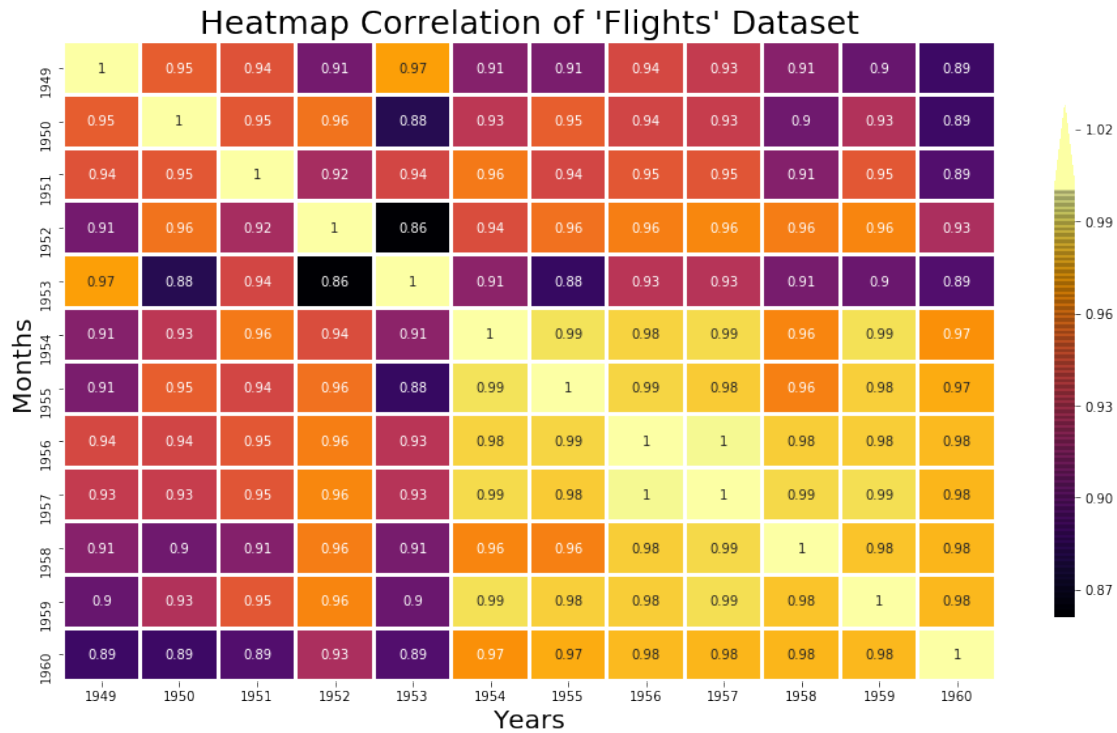
```
# create heatmap seaborn
```

```
cbar_kws = {"shrink":.8,
            'extend':'max',
            'extendfrac':.2,
            "drawedges":True}
```

```
sns.heatmap(flights_df.corr(), cmap="inferno", annot = True, linewidth
```

```
= 2, cbar_kws=cbar_kws)
```

```
plt.title("Heatmap Correlation of 'Flights' Dataset", fontsize = 25)
plt.xlabel("Years", fontsize = 20)
plt.ylabel("Months", fontsize = 20)
plt.show()
```



```
print("Thank you ..... -:)"
```

Thank you -:)