Explore Weather Trends

July25,2019

Picture Reference:https://indianexpress.com/article/india/indore-indias-cleanest-city-swachh-bharat-mission-5662774/ (https://indianexpress.com/article/india/indore-indias-cleanest-city-swachh-bharat-mission-5662774/)

Udacity - Data Analyst Nanodegree

Project -1, Explore Data Trends

Submitted By: Shreyas Shukla

OVERVIEW:

In this project, I have analyzed mean local temperature of Indore, INDIA (22.7196° N, 75.8577° E) and mean global temperature since 1850. I had been provided by Udacity with a database from where I extracted the data relevant to this project, using SQL. This project begins from the extracted files from the database.

GOALS:

1. Extraction of data from database using SQL and export as CSV file
2. Make a chart/graph visualisation
3. Observation and Inference based on graphs

TOOLS USED:

1. SQL
2. Python
3. ANACONDA - Jupyter Notebook
4. Google Sheets

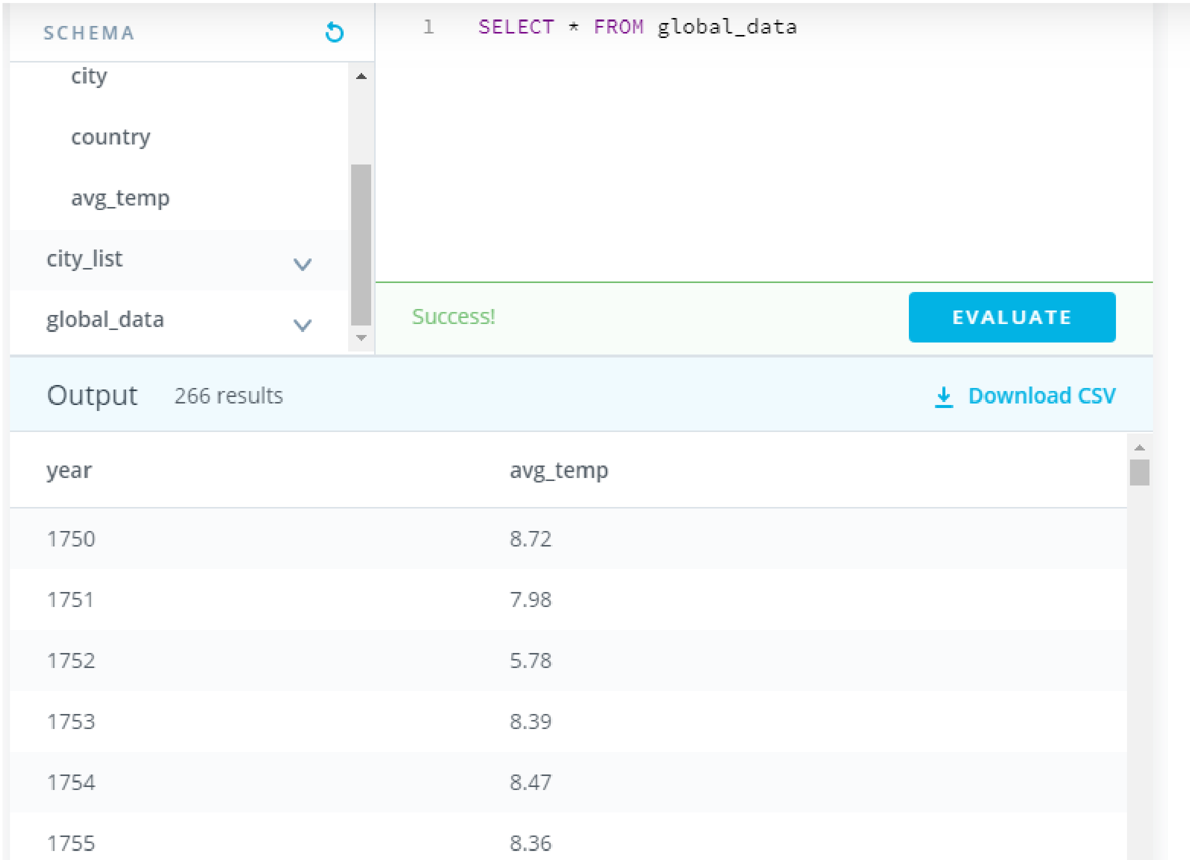
STEPS:

Extract data from database using SQL

In [1]:
```
from IPython.core.display import Image
```

In [2]:
```
Image(filename= 'C:/Users/shrey/Desktop/Global.png')
```

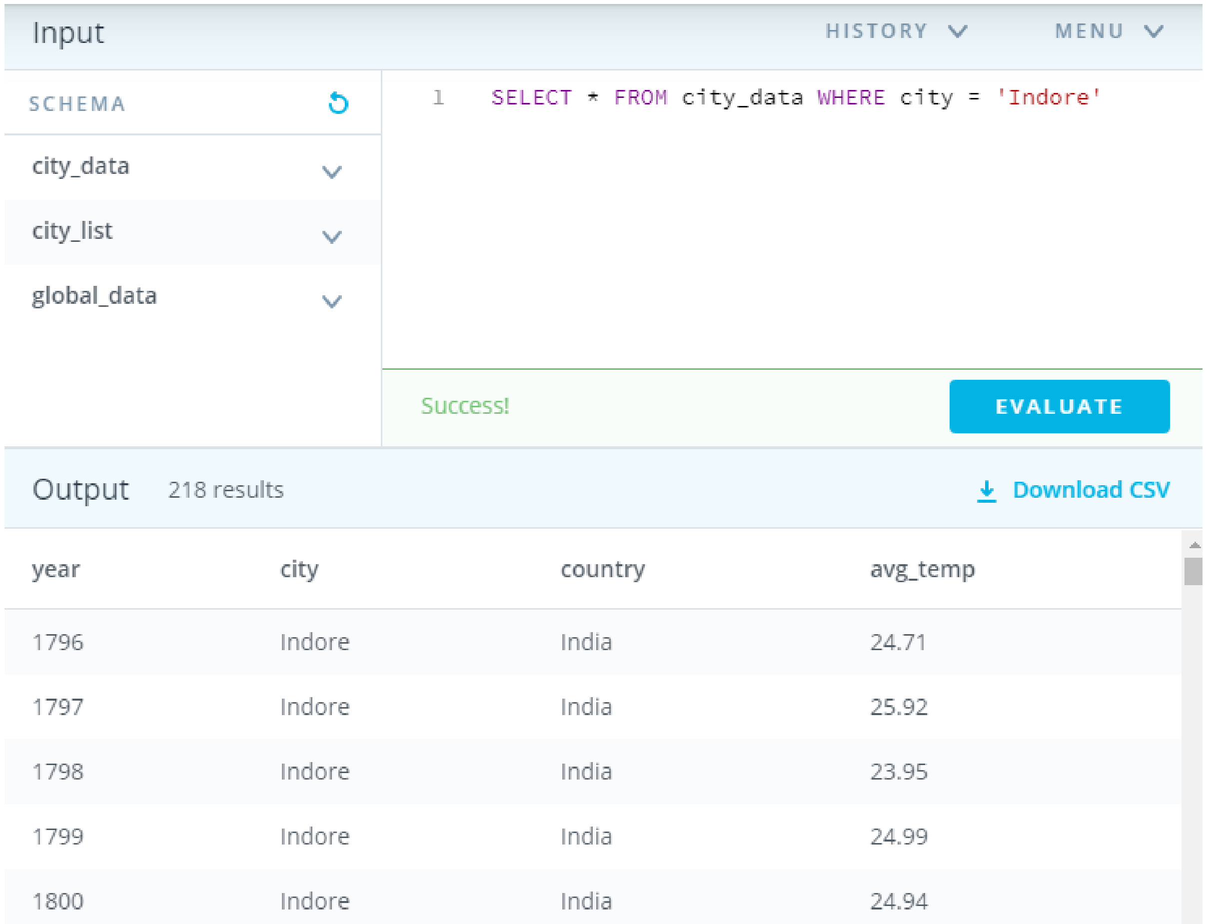
Out[2]:



year	avg_temp
1750	8.72
1751	7.98
1752	5.78
1753	8.39
1754	8.47
1755	8.36

In [3]:
```
Image(filename= 'C:/Users/shrey/Desktop/Indore.png')
```

Out[3]:



Import required python libraries

In [4]:
```
import pandas as pd
import numpy as np
import seaborn as sns
import matplotlib.pyplot as plt
```

Read extracted csv files of Global mean temperature and that of Indore city. Get maximum and minimum temperatures from both the datasets

In [5]:
```
Global = pd.read_csv('C:/Users/shrey/Desktop/UDACITY/Project1/Global.csv')
```

In [6]:
```
Indore = pd.read_csv('C:/Users/shrey/Desktop/UDACITY/Project1/Indore.csv')
```

In [7]:
```
Indore.head(5)
```

Out[7]:

	year	city	country	avg_temp
0	1796	Indore	India	24.71
1	1797	Indore	India	25.92
2	1798	Indore	India	23.95
3	1799	Indore	India	24.99
4	1800	Indore	India	24.94

In [8]:
```
Indore.max()
```

Out[8]:
```
year          2013
city        Indore
country      India
avg_temp     26.41
dtype: object
```

In [9]:
```
Indore.min()
```

Out[9]:
```
year          1796
city        Indore
country      India
avg_temp      19.6
dtype: object
```

In [10]:
```
Global.head(5)
```

Out[10]:

	year	avg_temp
0	1750	8.72
1	1751	7.98
2	1752	5.78
3	1753	8.39
4	1754	8.47

In [11]:
```
Global.min()
```

Out[11]:
```
year        1750.00
avg_temp       5.78
dtype: float64
```

In [12]:
```
Global.max()
```

Out[12]:
```
year        2015.00
avg_temp       9.83
dtype: float64
```

As we can see, data for Indore before 1796 is not available. So, let us consider the Global data from 1796 onwards so as to make both the datasets compatible.

In [13]:
```
Global = Global[Global['year'] > 1795]
```

In [14]:
```
Global.head(5)
```

Out[14]:

	year	avg_temp
46	1796	8.27
47	1797	8.51
48	1798	8.67
49	1799	8.51
50	1800	8.48

Columns- 'city' and 'country' are not relevant for our analysis.

In [15]:
```
Indore.drop(["city","country"], axis = 1, inplace = True)
```

We have a common column 'Year' in both the datasets. So, Let's now merge the two datasets inorder to further help in our analysis. But as we can see both the datasets have the average temperatures under the same column name - "avg_temp". Thus it is required to rename these columns under different heads and then merge the two datasets

In [16]:
```
Global.rename(columns = {"avg_temp": "G_avg_temp"}, inplace = True)
```

In [17]:
```
Indore.rename(columns = {"avg_temp": "I_avg_temp"}, inplace = True)
```

In [18]:
```
common = pd.merge(Global,Indore, on='year', how='inner')
```

In [19]:
```
common = common.reindex()
```

In [20]:
```
common.index += 1
```

In [21]:
```
common.head(3)
```

Out[21]:

	year	G_avg_temp	I_avg_temp
1	1796	8.27	24.71
2	1797	8.51	25.92
3	1798	8.67	23.95

Check the average temperatures for missing values

In [22]:
```
len(common[common['G_avg_temp'].isnull()])
```

Out[22]: 0

In [23]:
```
len(common[common['I_avg_temp'].isnull()])
```

Out[23]: 7

We have found 7 missing values in mean temperature of Indore city. Let us fill these missing values using interpolation method.

In [24]:
```
common[common['I_avg_temp'].isnull()]
```

Out[24]:

	year	G_avg_temp	I_avg_temp
13	1808	7.63	NaN
14	1809	7.08	NaN
15	1810	6.92	NaN
16	1811	6.86	NaN
17	1812	7.05	NaN
68	1863	8.11	NaN
69	1864	7.98	NaN

In [25]:
```
common['I_avg_temp'] = common['I_avg_temp'].interpolate(method ='linear', limi
t_direction ='forward')
```

In [26]:
```
len(common[common['I_avg_temp'].isnull()])
```

Out[26]: 0

Let us calculate Moving Averages (window = 40 years) of average temperatures for Global and Indore under the columns "MA-G" and "MA-I" respectively

In [27]:
```
common['MA-I'] = common.I_avg_temp.rolling(41).mean()
```

In [28]:
```
common['MA-G'] = common.G_avg_temp.rolling(41).mean()
```

In [29]:
```
common
```

Out[29]:

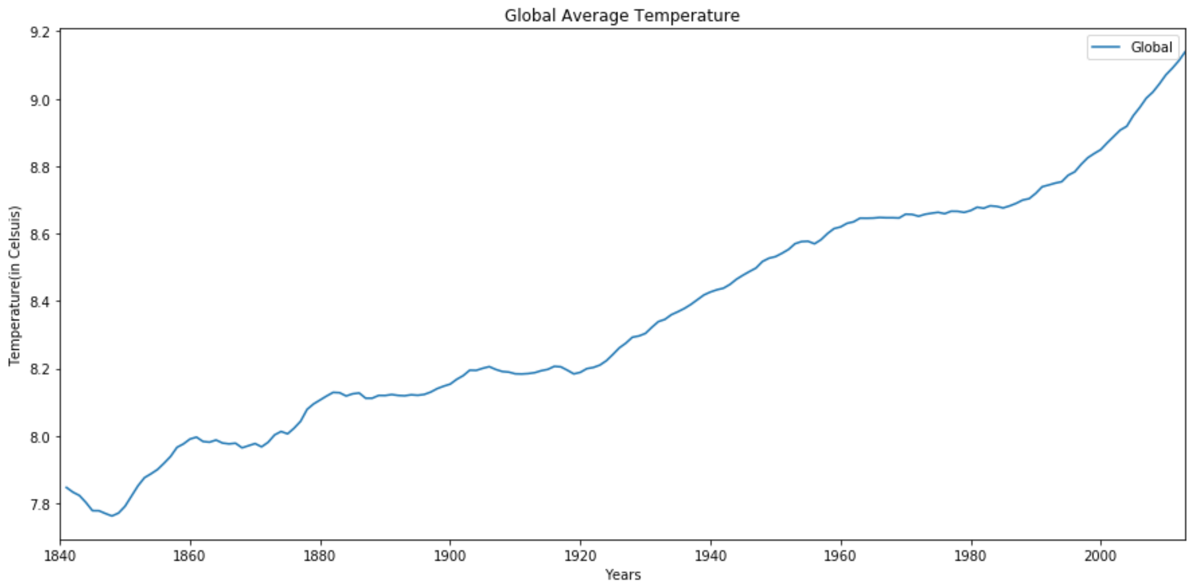
	year	G_avg_temp	l_avg_temp	MA-l	MA-G
1	1796	8.27	24.71	NaN	NaN
2	1797	8.51	25.92	NaN	NaN
3	1798	8.67	23.95	NaN	NaN
4	1799	8.51	24.99	NaN	NaN
5	1800	8.48	24.94	NaN	NaN
6	1801	8.59	23.86	NaN	NaN
7	1802	8.58	25.41	NaN	NaN
8	1803	8.50	25.17	NaN	NaN
9	1804	8.84	25.51	NaN	NaN
10	1805	8.56	25.06	NaN	NaN
11	1806	8.43	24.96	NaN	NaN
12	1807	8.28	24.36	NaN	NaN
13	1808	7.63	24.35	NaN	NaN
14	1809	7.08	24.34	NaN	NaN
15	1810	6.92	24.33	NaN	NaN
16	1811	6.86	24.32	NaN	NaN
17	1812	7.05	24.31	NaN	NaN
18	1813	7.74	24.30	NaN	NaN
19	1814	7.59	23.50	NaN	NaN
20	1815	7.24	23.84	NaN	NaN
21	1816	6.94	23.44	NaN	NaN
22	1817	6.98	23.62	NaN	NaN
23	1818	7.83	24.04	NaN	NaN
24	1819	7.37	23.71	NaN	NaN
25	1820	7.62	23.89	NaN	NaN
26	1821	8.09	24.55	NaN	NaN
27	1822	8.19	24.61	NaN	NaN
28	1823	7.72	24.48	NaN	NaN
29	1824	8.55	25.08	NaN	NaN
30	1825	8.39	24.83	NaN	NaN
...	...	...	...	...	...
189	1984	8.69	25.09	25.149024	8.680488
190	1985	8.66	25.53	25.168537	8.675854
191	1986	8.83	25.34	25.191951	8.681951
192	1987	8.99	26.02	25.213171	8.689512

	year	G_avg_temp	l_avg_temp	MA-l	MA-G
193	1988	9.20	25.89	25.231220	8.699268
194	1989	8.92	25.28	25.230976	8.703415
195	1990	9.23	25.16	25.224634	8.719024
196	1991	9.18	25.40	25.244390	8.738780
197	1992	8.84	25.41	25.245366	8.743902
198	1993	8.87	25.50	25.242927	8.749512
199	1994	9.04	25.01	25.225366	8.753659
200	1995	9.35	25.50	25.232195	8.772927
201	1996	9.04	25.55	25.247805	8.782927
202	1997	9.20	24.77	25.246585	8.805366
203	1998	9.52	25.87	25.266585	8.824634
204	1999	9.29	25.50	25.261220	8.837317
205	2000	9.20	25.62	25.270732	8.848780
206	2001	9.41	25.59	25.279024	8.869024
207	2002	9.57	26.15	25.314146	8.887805
208	2003	9.53	25.82	25.336585	8.906829
209	2004	9.32	25.98	25.355854	8.918049
210	2005	9.70	25.40	25.364146	8.949512
211	2006	9.53	25.78	25.374390	8.973902
212	2007	9.73	25.66	25.378049	9.001463
213	2008	9.43	25.30	25.388293	9.019268
214	2009	9.51	26.41	25.429024	9.043415
215	2010	9.70	26.31	25.446341	9.070244
216	2011	9.52	25.45	25.457073	9.090244
217	2012	9.51	25.39	25.478537	9.112439
218	2013	9.61	25.94	25.495122	9.139512

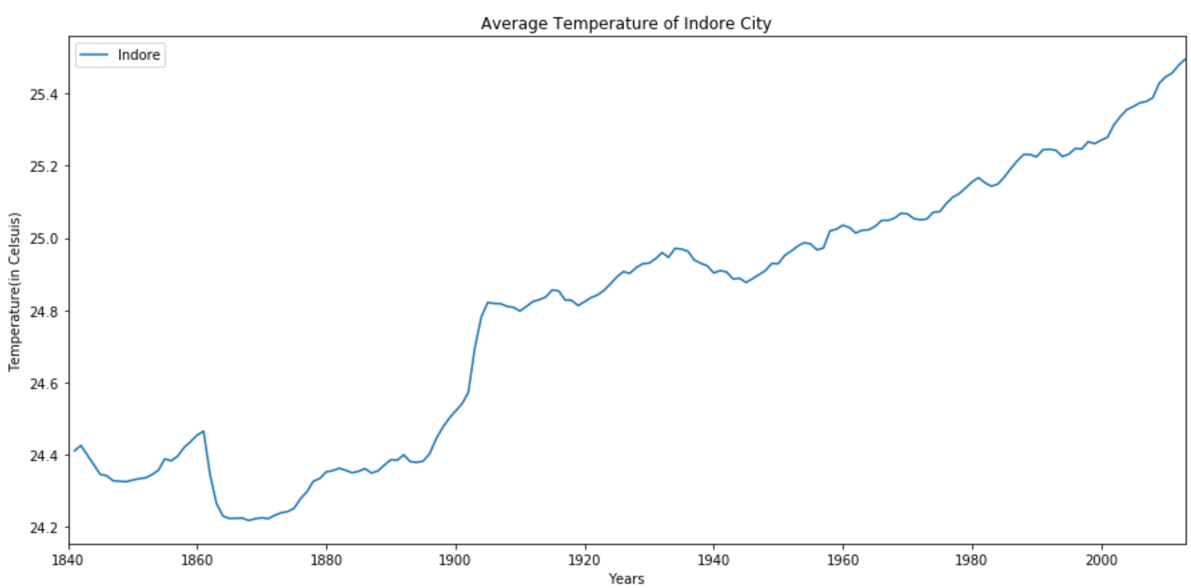
218 rows × 5 columns

Since the window is taken as 40, first 40 moving average values will be NaN. Thus, for further analysis using moving average, we have to consider the records from 1840 onwards. (Note: One can take any value of window. Larger the window, lesser the noise in data)

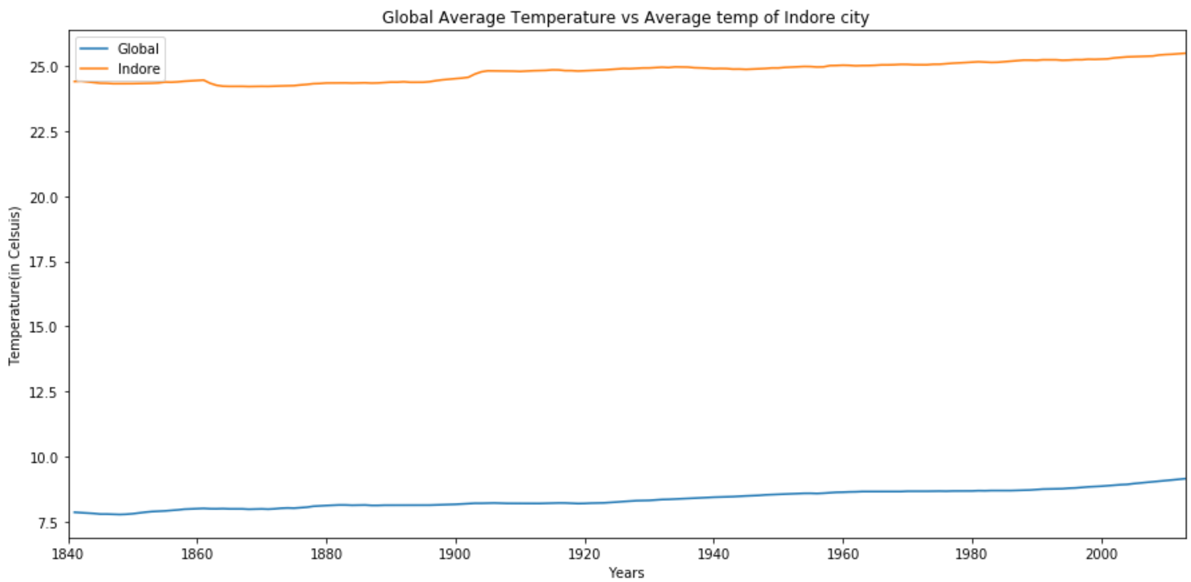
In [30]:
```
plt.figure(figsize=(15,7))
plt.plot('year','MA-G',data = common[common["year"]>1840], label = 'Global')
plt.legend()
plt.xlabel("Years")
plt.ylabel("Temperature(in Celsuis)")
plt.title("Global Average Temperature")
plt.xlim(1840,2013)
plt.show()
```



In [31]:
```
plt.figure(figsize=(15,7))
plt.plot('year','MA-I',data = common[common["year"]>1840], label = 'Indore')
plt.legend()
plt.xlabel("Years")
plt.ylabel("Temperature(in Celsuis)")
plt.title("Average Temperature of Indore City")
plt.xlim(1840,2013)
plt.show()
```



In [47]:
```
plt.figure(figsize=(15,7))
plt.plot('year','MA-G',data = common[common["year"]>1840], label = 'Global')
plt.plot('year','MA-I',data = common[common["year"]>1840], label = 'Indore')
plt.legend()
plt.xlabel("Years")
plt.ylabel("Temperature(in Celsuis)")
plt.title("Global Average Temperature vs Average temp of Indore city")
plt.xlim(1840,2013)
plt.show()
```



OBSERVATIONS:

1. Very big diffference in the average temperature of Indore and that of world. This is mostly due to the fact that city of Indore lies much closer to the equator.
   A. Global average temperature is rising constantly since last one and half century.
   B. Graph 1 shows declining global average temperature before 1860 which goes in line with the 'Mini Ice Age' theory.
   C. Global min. average temperature was encountered in 1752: 5.78 degree Celsuis while maximum average temperature was seen in 2015: 9.83 degree Celsuis
   D. Not only the average temperature is increasing, but rate of its increase is also increasing.

Now let us compare the temperature rise in the major cities around the world. We've considered Moving Average window as 10 years so as to better understand the variation in average temperature at various places. Also, instead of interpolation method, we're using mean of all the average temperatures to fill the missing values.

In [33]:
```
Antananarivo = pd.read_csv('C:/Users/shrey/Desktop/UDACITY/Project1/Antananarivo.csv')
```

In [34]:
```
Cordoba = pd.read_csv('C:/Users/shrey/Desktop/UDACITY/Project1/Cordoba.csv')
```

In [35]:
```
LA = pd.read_csv('C:/Users/shrey/Desktop/UDACITY/Project1/LA.csv')
```

In [36]:
```
London = pd.read_csv('C:/Users/shrey/Desktop/UDACITY/Project1/London.csv')
```

In [37]:
```
Ottawa = pd.read_csv('C:/Users/shrey/Desktop/UDACITY/Project1/Ottawa.csv')
```

In [38]:
```
Perth = pd.read_csv('C:/Users/shrey/Desktop/UDACITY/Project1/Perth.csv')
```

In [39]:
```
Tokyo = pd.read_csv('C:/Users/shrey/Desktop/UDACITY/Project1/Tokyo.csv')
```

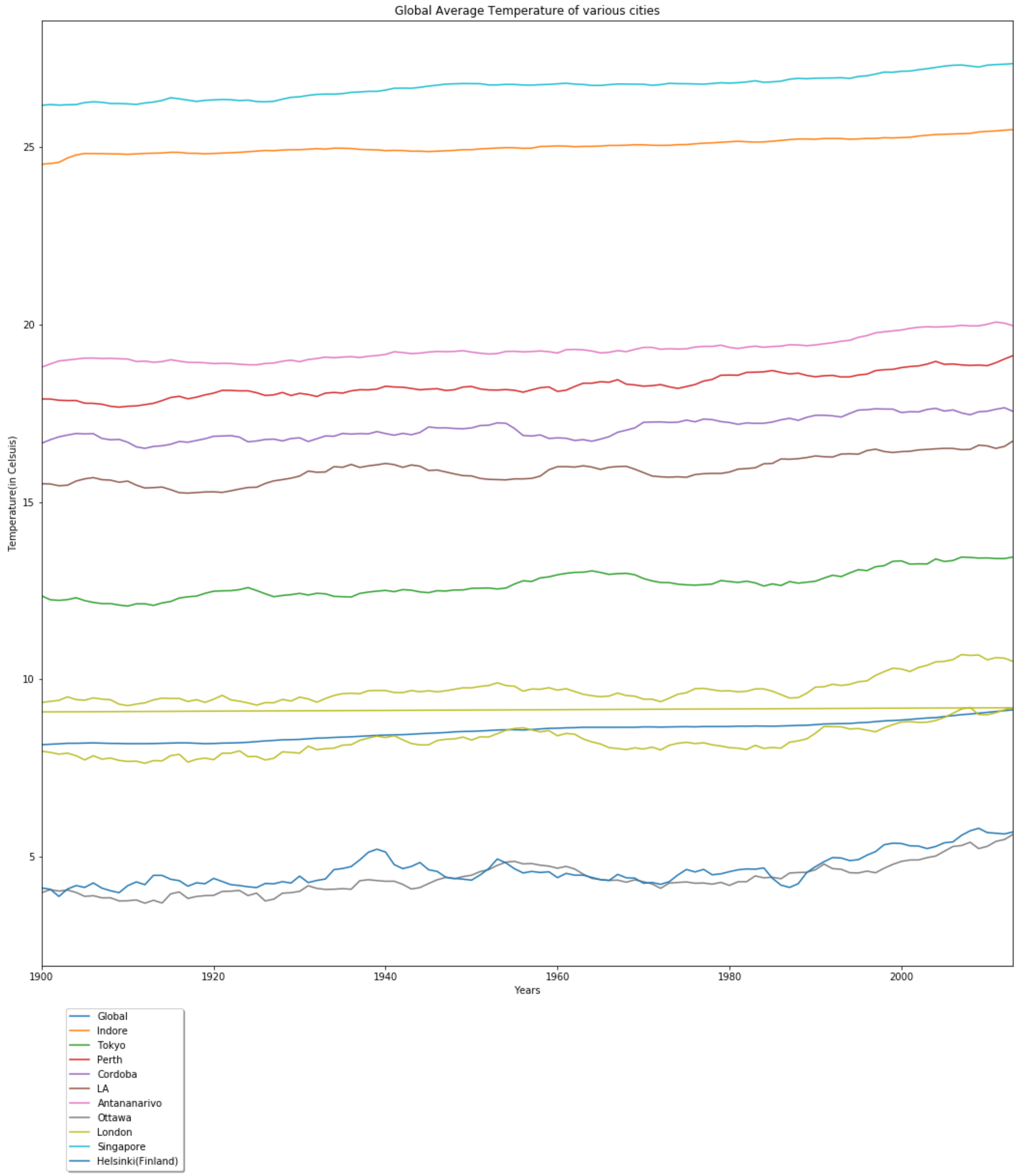
In [40]:
```
Singapore = pd.read_csv('C:/Users/shrey/Desktop/UDACITY/Project1/Singapore.cs
v')
```

In [41]:
```
Helsinki = pd.read_csv('C:/Users/shrey/Desktop/UDACITY/Project1/Helsinki.csv')
```

In [42]:
```
London = London.fillna(London['avg_temp'].mean())
Ottawa = Ottawa.fillna(Ottawa['avg_temp'].mean())
LA = LA.fillna(LA['avg_temp'].mean())
Perth = Perth.fillna(Perth['avg_temp'].mean())
Antananarivo = Antananarivo.fillna(Antananarivo['avg_temp'].mean())
Cordoba = Cordoba.fillna(Cordoba['avg_temp'].mean())
Tokyo = Tokyo.fillna(Tokyo['avg_temp'].mean())
Singapore = Singapore.fillna(Singapore['avg_temp'].mean())
Helsinki = Helsinki.fillna(Helsinki['avg_temp'].mean())
```

In [45]:
```
Tokyo['MA-Tokyo'] = Tokyo.avg_temp.rolling(11).mean()
Cordoba['MA-Cordoba'] = Cordoba.avg_temp.rolling(11).mean()
Ottawa['MA-Ottawa'] = Ottawa.avg_temp.rolling(11).mean()
Antananarivo['MA-Antananarivo'] = Antananarivo.avg_temp.rolling(11).mean()
Perth['MA-Perth'] = Perth.avg_temp.rolling(11).mean()
LA['MA-LA'] = LA.avg_temp.rolling(11).mean()
London['MA-London'] = London.avg_temp.rolling(11).mean()
Singapore['MA-Singapore'] = Singapore.avg_temp.rolling(11).mean()
Helsinki['MA-Helsinki'] = Helsinki.avg_temp.rolling(11).mean()
```

In [46]:
```
plt.figure(figsize=(18,18))
plt.plot('year','MA-G',data = common[common["year"]>1845], label = 'Global')
plt.plot('year','MA-I',data = common[common["year"]>1845], label = 'Indore')
plt.plot('year','MA-Tokyo',data = Tokyo[Tokyo["year"]>1845], label = 'Tokyo')
plt.plot('year','MA-Perth',data = Perth[Perth["year"]>1845], label = 'Perth')
plt.plot('year','MA-Cordoba',data = Cordoba[Cordoba["year"]>1845], label = 'Co
rdoba')
plt.plot('year','MA-LA',data = LA[LA["year"]>1845], label = 'LA')
plt.plot('year','MA-Antananarivo',data = Antananarivo[Antananarivo["year"]>184
5], label = 'Antananarivo')
plt.plot('year','MA-Ottawa',data = Ottawa[Ottawa["year"]>1845], label = 'Ottaw
a')
plt.plot('year','MA-London',data = London[London["year"]>1845], label = 'Londo
n')
plt.plot('year','MA-Singapore',data = Singapore[Singapore["year"]>1845], label
= 'Singapore')
plt.plot('year','MA-Helsinki',data = Helsinki[Helsinki["year"]>1845], label =
'Helsinki(Finland)')
plt.legend( bbox_to_anchor=(0.15, -0.04),fancybox=True, shadow=True)
plt.xlabel("Years")
plt.ylabel("Temperature(in Celsuis)")
plt.title("Global Average Temperature of various cities")
plt.xlim(1900,2013)
plt.show()
```



OBSERVATION:

We can see that cities further towards the pole have much noisier curve. Thus, it can be concluded that cities/places away from equator are facing more average temperature variations than those near the equator.