Project 1

Pragati Mehra

Step 1

SQL Query used to extract the data

Select * from city_data where city='Delhi' and co;

Select * from global_data;

Step 2

Tools used

SQL to extract the Data

Python as language to right the code

Spyder as environment

Libraries like Numpy and Pandas to find the average temperature and

Step 3

How to calculate moving

Using function .rolling function in the avg_temp

Window has to be described the number of terms to be averaged at one go

CODE

```
x["rollingAverage"] = x["avg_temp"].rolling(window = avg_num).mean()
x.reset_index(inplace = True, drop = True)
b=x["rollingAverage"]
```

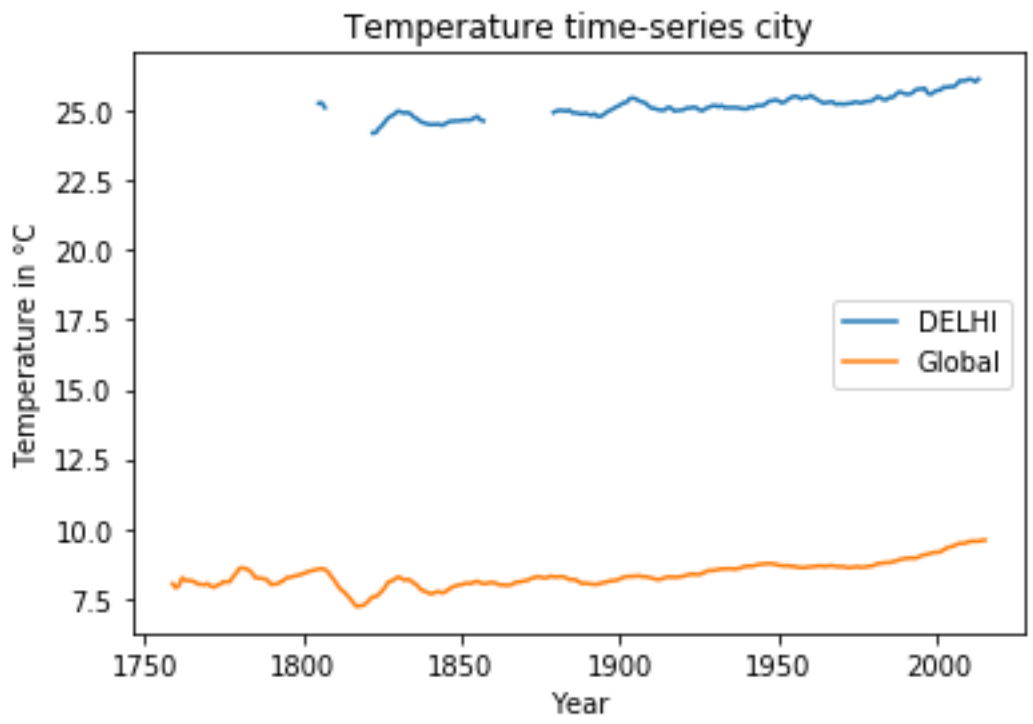
Step 4

Key consideration

1 Average trends vs rolling Average in global and city temperatures

2 Rolling average in particular city vs the global

Line chart



Printing city avg with respect to global avg

CODE

```
import pandas as pd
import numpy as np
import matplotlib.pyplot as plt

x = pd.read_csv('global_delhi.csv')
y=  pd.read_csv('global_city.csv')
avg_num = 10
x["rollingAverage"] = x["avg_temp"].rolling(window = avg_num).mean()
y["rollingAverage"] = y["avg_temp"].rolling(window = avg_num).mean()
```

```
print("Printing city avg with respect to global avg")


plt.plot("year","rollingAverage", data =x, label = "DELHI")
plt.plot("year","rollingAverage", data =y, label = "Global")
plt.legend()
plt.ylabel("Temperature in °C")
plt.xlabel("Year")
plt.title("Temperature time-series city")
plt.show()plt.ylabel("Temperature in °C")
plt.xlabel("Year")
plt.title("Temperature time-series city")
plt.tight_layout()
plt.show()
```

Observations

1 At some Year corresponding value of the Temp is missing for Delhi

2 Between 1800 and 1850 the Rolling average pattern of both city as well as the Global is Same as in the drop and rise of the Graph is pretty similar

3 from 1900 Till 2000 the pattern has a peak in city and then show a very similar pattern as Global pattern

4 The average of a city in general more than the average of global