**Visualization Lab Part Two by Group 19**

**1-In each of the source codes, there is a comment and corresponding code about “Preparing data”. Find each one of them and explain what each one does as far as data preprocessing before drawing the figures.**

Preparing data Code of Bar chart:

# Preparing data

data = [go.Bar(x=new\_df['State'], y=new\_df['Confirmed'])]

This line of code sets the x-axis for the bar chart are states names, sets the y-axis for the bar chart are number of confirmed cases.

Preparing data Code for stack bar chart:

# Preparing data

trace1 = go.Bar(x=new\_df['Country'], y=new\_df['Unrecovered'], name= 'Unrecovered',

marker={'color': '#CD7F32'})

trace2 = go.Bar(x=new\_df['Country'], y=new\_df['Recovered'], name= 'Recovered',

marker={'color': '#9EA0A1'})

trace3 = go.Bar(x=new\_df['Country'], y=new\_df['Deaths'], name= 'Deaths',

marker={'color': '#FFD700'})

data = [trace1, trace2, trace3]

These lines of codes set three stack. They have the same x-axis ‘Country’. The first one’s y-axis is ‘Unrecovered’ and it represents the number of unrecovered cases, using a color code ‘#CD7F32’.

The second one’s y-axis is ‘Recovered’ and it represents the number of recovered cases, using a color code ‘#9EA0A1’.

The third one’s y-axis is ‘Deaths’ and it represents the number of deaths cases, using a color code ‘#FFD700’.

Preparing data code for line chart:

# Preparing data

data = [go.Scatter(x=df['Date'], y=df['Confirmed'], mode= 'lines', name= 'Death')]

This line of code sets the mode as ‘lines’ , x-axis as date and y-axis as number of confirmed cases.

Preparing data code for multiline chart:

# Preparing data

trace1 = go.Scatter(x=df['Date'], y=df['Death'], mode='lines', name='Death')

trace2 = go.Scatter(x=df['Date'], y=df['Recovered'], mode='lines', name='Recovered')

trace3 = go.Scatter(x=df['Date'], y=df['Unrecovered'], mode='lines', name='Unrecovered')

data = [trace1,trace2,trace3]

These lines of codes set three lines for the multiline chart. X-axis represents data for all the three lines. The first line’s y-axis represents death cases; the second line’s y=axis represents recovered cases; the third line’s y-axis represents unrecovered cases.

Preparing data code for bubble chart:

data = [

go.Scatter(x=new\_df['Recovered'],

y=new\_df['Unrecovered'],

text=new\_df['Country'],

mode='markers',

marker=dict(size=new\_df['Confirmed']/100,

color=new\_df['Confirmed']/100, showscale=True))

]

These lines of codes set x-axis as recovered cases, y-axis as unrecovered cases.

Preparing data code for heatmap:

# Preparing data

data = [go.Heatmap(x=df['Day'],

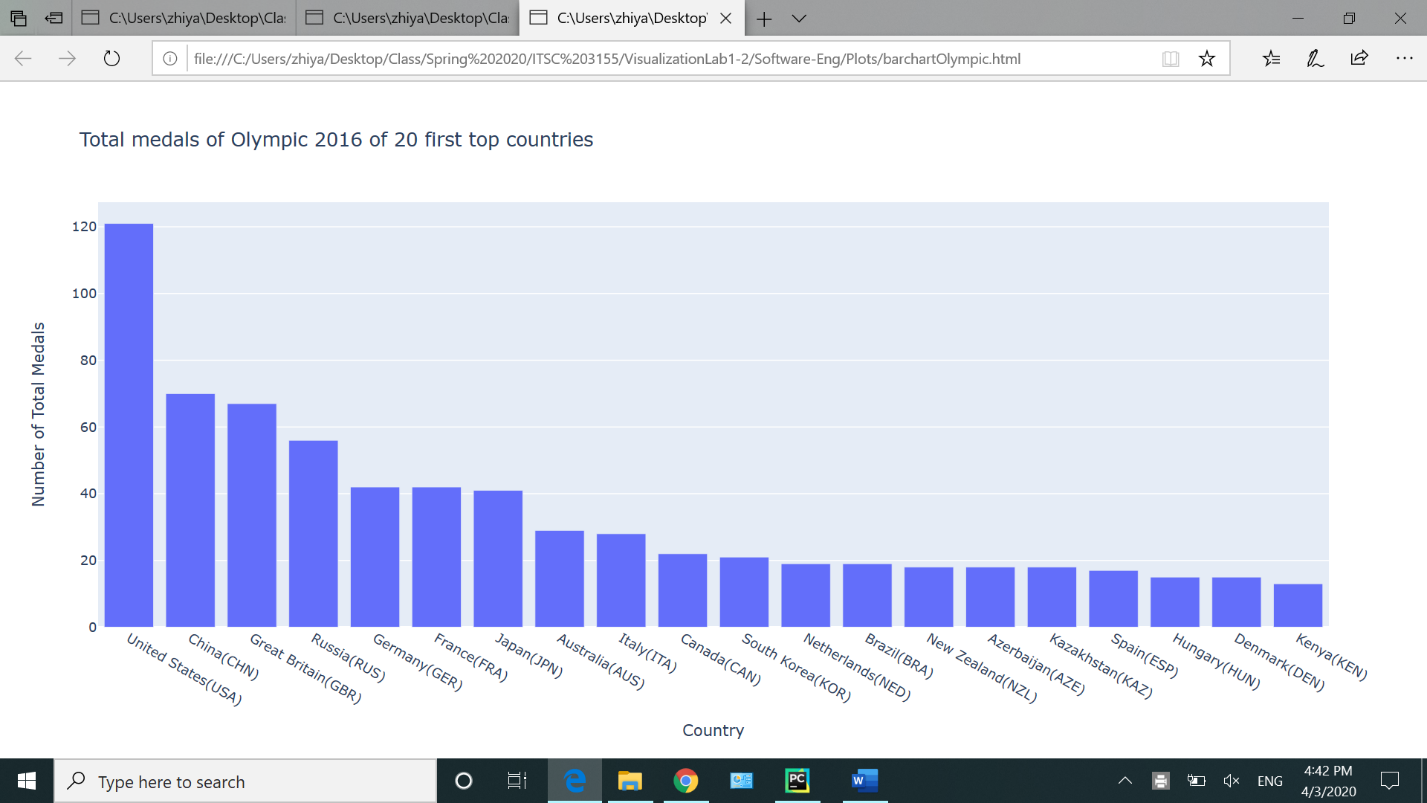
y=df['WeekofMonth'],

z=df['Recovered'].values.tolist(),

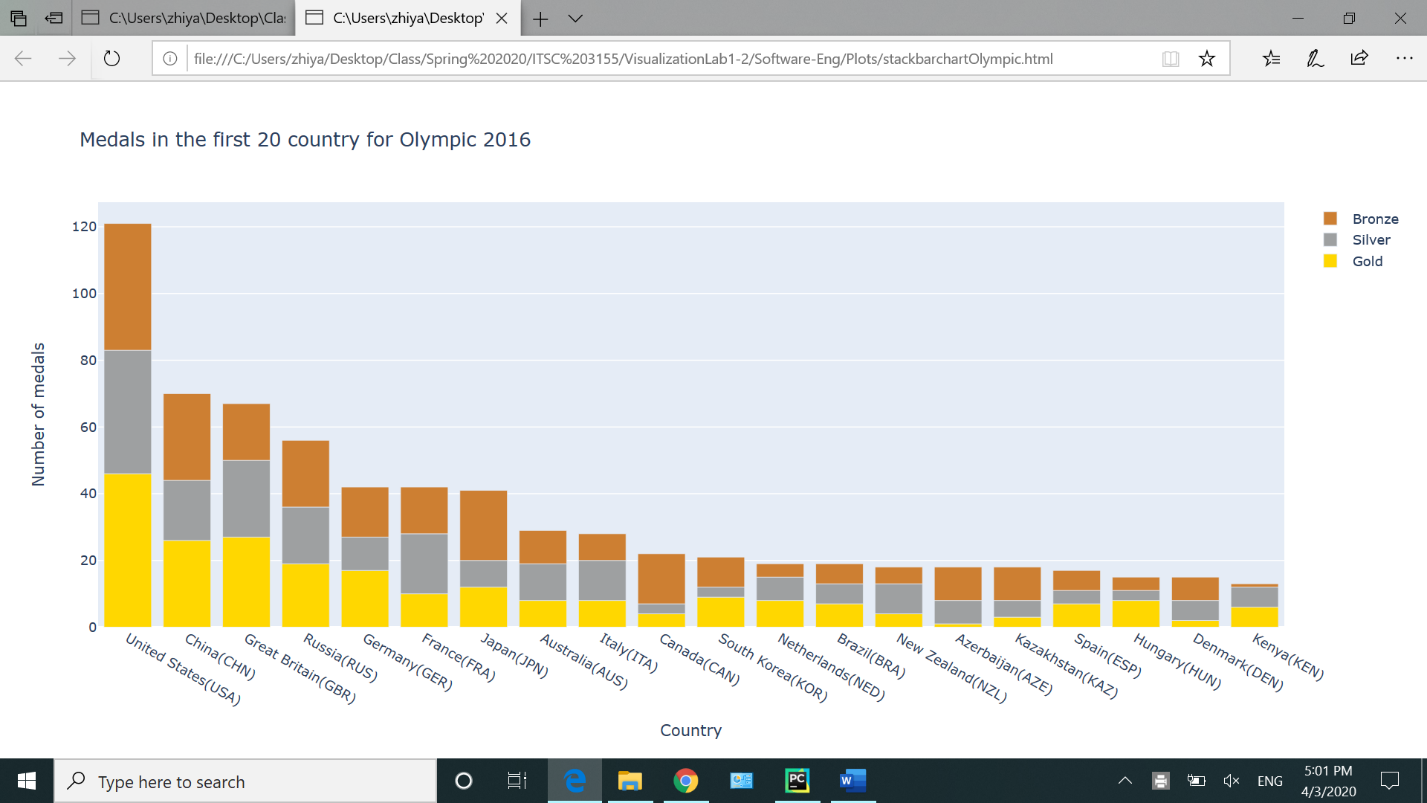
colorscale= 'Jet')]

Set x-axis as day, y-axis as week of month, color scale as jet, z as recovered.

**2- Draw a bar chart to represent the total medals of Olympic 2016 of 20 first top countries. (use Olympic2016Rio.csv file)**



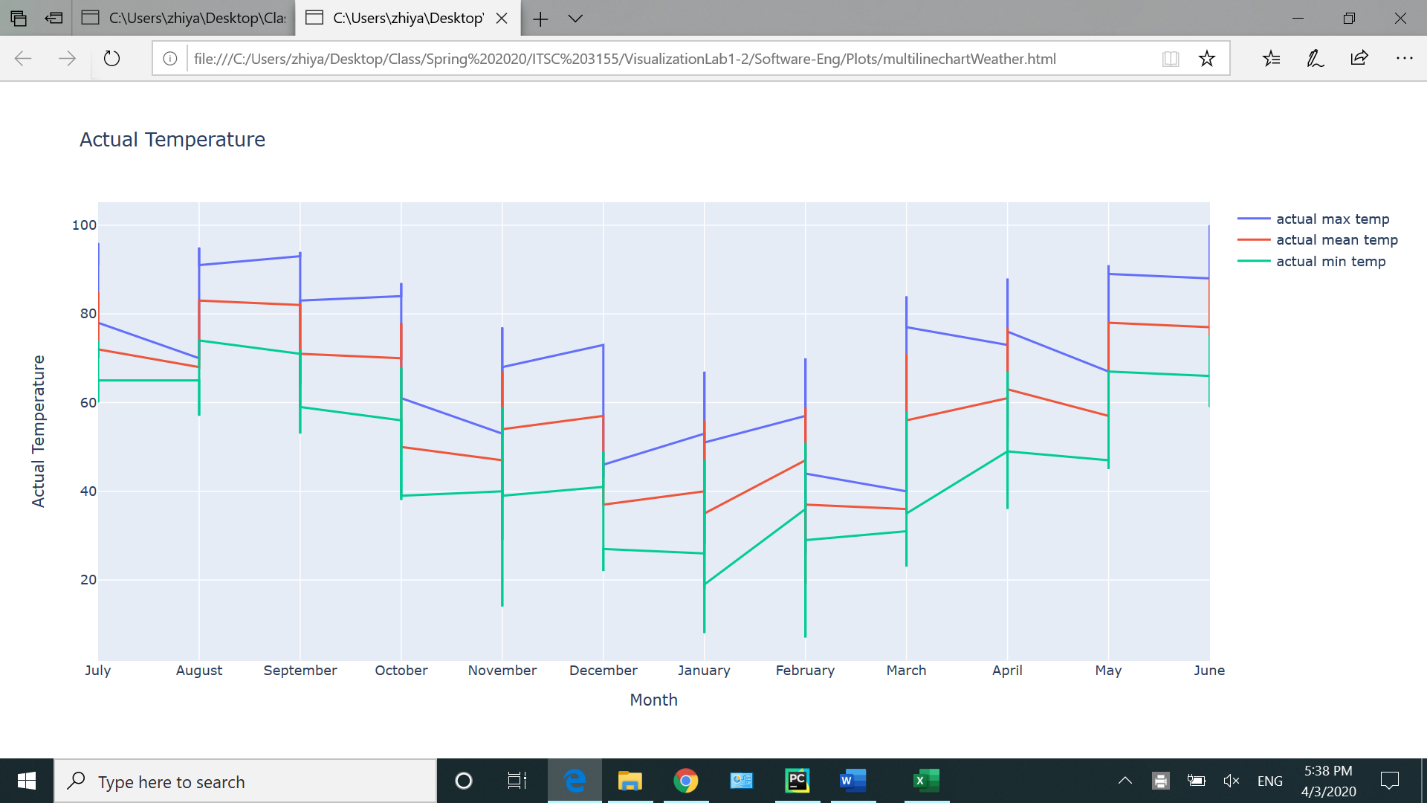
**3- Draw a stack bar chart to represent the Gold, Silver, Bronze medals of Olympic 2016 of 20 first top countries. (use Olympic2016Rio.csv file)**



**4- Draw a line chart to represent the actual max temperature of each month in weather statistics. (use Weather2014-15.csv file)**



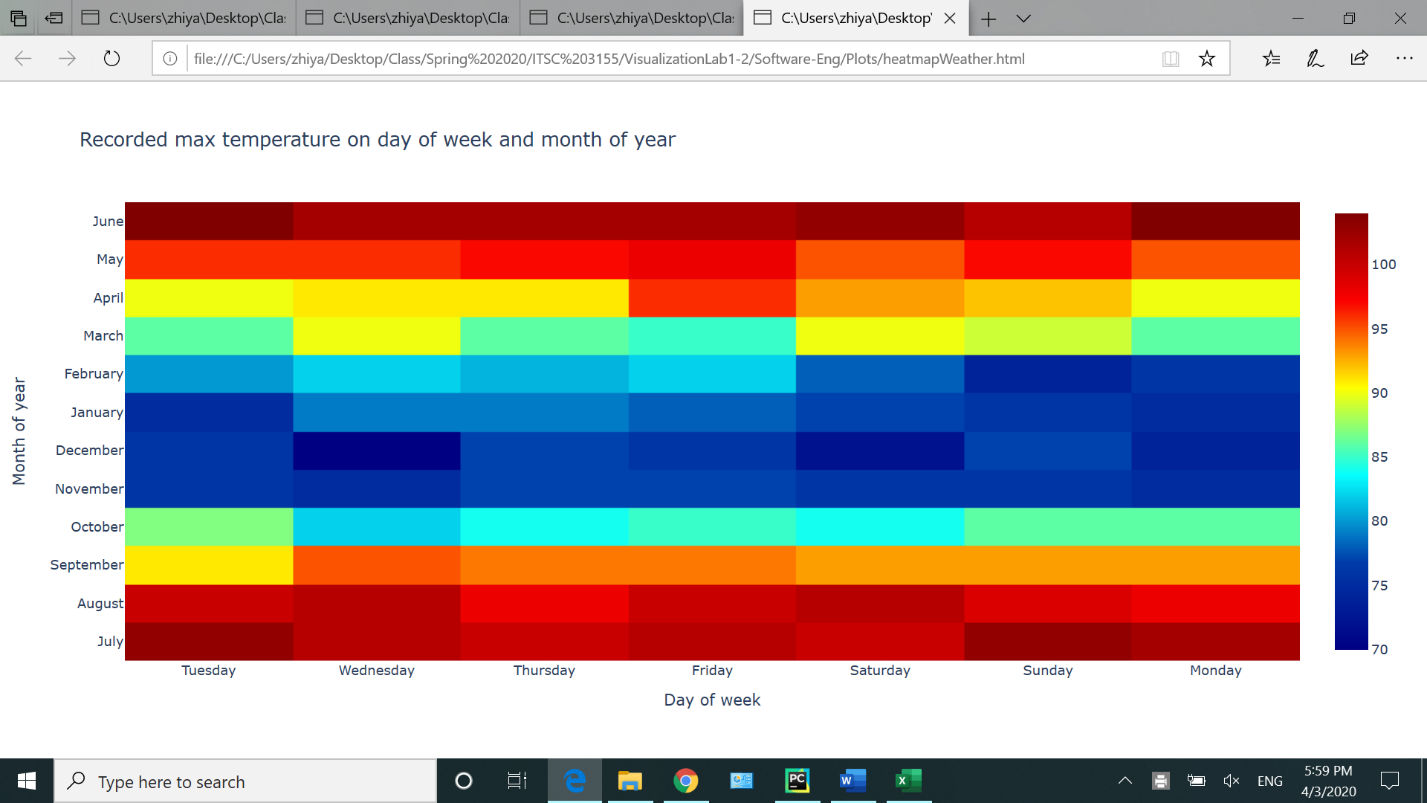
**5- Draw a multi line chart to represent the actual max, min and mean temperature of each month in weather statistics. (use Weather2014-15.csv file)**



**6- Draw a bubble chart to represent the average min and max temperature of each month in weather statistics. (use Weather2014-15.csv file)**



**7- Draw a heatmap to represent the recorded max temperature on day of week and month of year. (use Weather2014-15.csv file)**



**8- Reflect on the creation of these graphs in python as far as complexity of the creation process, clarity of the figures, and usefulness as far as providing information.**

All the provided information are very useful.