

Plotly Express Documentation

Step 1 - Installing **plotly** module. You can do it inside Jupyter Notebook as shown below

Plotly installation

```
In [3]: ! pip install plotly

Collecting plotly
  Downloading plotly-4.6.0-py2.py3-none-any.whl (7.1 MB)
Collecting retrying>=1.3.3
  Downloading retrying-1.3.3.tar.gz (10 kB)
Requirement already satisfied: six in c:\programdata\anaconda3\lib\site-packages (from plotly) (1.14.0)
Building wheels for collected packages: retrying
  Building wheel for retrying (setup.py): started
  Building wheel for retrying (setup.py): finished with status 'done'
  Created wheel for retrying: filename=retrying-1.3.3-py3-none-any.whl size=11435 sha256=cc9445bfceefa56365d44987adf6818541ce4ad277b55e7c91234a370fb5dd7d
  Stored in directory: c:\users\kanav\appdata\local\pip\cache\wheels\f9\8d\f6af3f7f9eea3553bc2fe6d53e4b287dad18b06a861ac56d
Successfully built retrying
Installing collected packages: retrying, plotly
Successfully installed plotly-4.6.0 retrying-1.3.3
```

Step 2 - Reading the csv data into a dataframe.

```
In [2]: # Importing the data

nyc = pd.read_csv('data/nyc_weather.csv')
```

Step 3 - Import required library - **plotly.express**

```
In [5]: # Importing required Library

import plotly.express as px
```

Step 4 - Scatter Plot using plotly.express

Note - Scatter Plot is a bivariate plot. Bivariate means it requires two variables / features / columns. You should make a note that both the variables should be real numerical valued.

```
In [6]: # Scatter Plot
px.scatter(nyc, x = 'Temperature', y = 'DewPoint')
```

Step 5 - Box Plot using plotly.express

Note - Box Plot can be used to create a univariate or bivariate plot. For a univariate box plot, the column type should be real numerical. For a bivariate box plot, one column should be categorical and another column should be real numerical. Below is an example of code for bivariate box plot.

```
In [10]: # Box Plot in Plotly Express
px.box(titanic, x = 'sex', y = 'age')
```

Step 6 - Pie Chart Plot using plotly.express

Note - Pie Chart Plot can be used to create a bivariate plot. For a bivariate pie chart plot, one column should be categorical and another column should be real numerical. Below is an example of code for the plot.

names: It should be categorical column

values: It should be numeric column

```
In [20]: px.pie(tips, names = 'day', values = 'total_bill')
```

Step 7 - Choropleth Plot using plotly.express

Note - Parameters for choropleth plot:

locations: It can be columns like - 'Country', 'Zip Code', etc...

color: It can be a column, value of which is used to assign color to marks

locationmode: It should be either one of 'ISO-3', 'USA-states', or 'country names'. Determines the set of locations used to match entries in **locations** to regions on the map.

```
In [28]: temp_df = group_df.get_group("2020-03-15")
temp_df.head()
```

Out[28]:	Country	Confirmed	Recovered	Deaths
9964	Afghanistan	16	0	0
9965	Albania	42	0	1
9966	Algeria	48	12	4
9967	Andorra	1	1	0
9968	Angola	0	0	0

```
In [29]: fig = px.choropleth(temp_df,  
                             locations='Country',  
                             color = 'Confirmed',  
                             hover_name='Confirmed',  
                             locationmode='country names')  
  
fig.show()
```

Step 8 - Animated Choropleth Plot using plotly.express

Note - Parameters for choropleth plot:

animation_frame: It should be a column like day, year, month, etc on which animation will be applied.

[illegible]