```
def configure_plotly_browser_state():
  import IPython
  display(IPython.core.display.HTML('''
        <script src="/static/components/requirejs/require.js"></script>
        <script>
          requirejs.config({
            paths: {
              base: '/static/base',
              plotly: 'https://cdn.plot.ly/plotly-1.5.1.min.js?noext',
            },
          });
        </script>
        '''))

configure_plotly_browser_state()
```

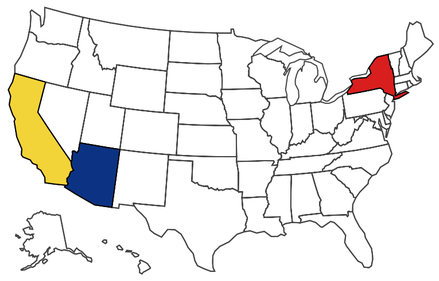
```
!pip install chart-studio
```

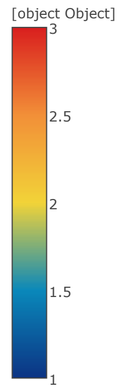
```
from chart_studio import plotly as py
import plotly.graph_objs as go
from plotly.offline import download_plotlyjs, init_notebook_mode, plot, iplot
init_notebook_mode(connected=True)
```

```
import pandas as pd
```

```
data=dict(type='choropleth',
          locations=['AZ','CA','NY'],
          locationmode='USA-states',
          colorscale='Portland',
          text=['text1','text2','text3'],
          z=[1.0,2.0,3.0],
          colorbar={'title':'Colorbar Title'})
```

```
configure_plotly_browser_state()
layout=dict(geo={'scope':'usa'})
choromap=go.Figure(data=[data],layout=layout)
iplot(choromap)
```






```
from google.colab import drive
drive.mount('/content/drive')
```

```
Mounted at /content/drive
```

```
df=pd.read_csv('/content/drive/MyDrive/Wharton Acads/Python/2011_US_AGRI_Exports')
```

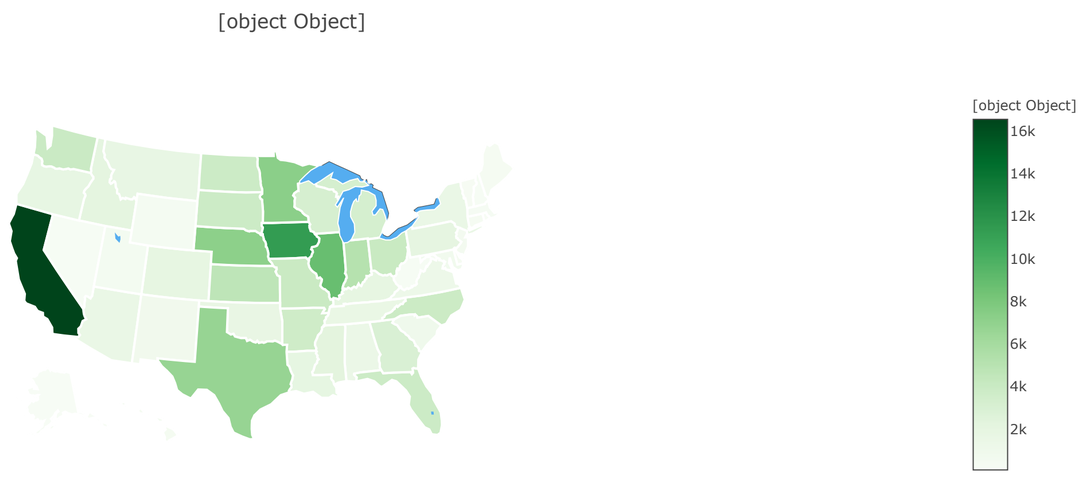
```
df.tail()
```

| | code | state | category | total exports | beef | pork | poultry | dairy | fruits fresh | fruits proc | total fruits | veggies fresh | veggies proc | total veggies | corn | wheat | cotton | text |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| **45** | VA | Virginia | state | 1146.48 | 39.5 | 16.9 | 164.7 | 47.85 | 11.7 | 24.8 | 36.48 | 10.4 | 16.9 | 27.25 | 39.5 | 77.5 | 64.84 | Virginia<br>Beef 39.5 Dairy 47.85<br>Fruits 36... |
| **46** | WA | Washington | state | 3894.81 | 59.2 | 0.0 | 35.6 | 154.18 | 555.6 | 1183.0 | 1738.57 | 138.7 | 225.1 | 363.79 | 29.5 | 786.3 | 0.00 | Washington<br>Beef 59.2 Dairy 154.18<br>Fruits... |
| **47** | WV | West Virginia | state | 138.89 | 12.0 | 0.3 | 45.4 | 3.90 | 3.7 | 7.9 | 11.54 | 0.0 | 0.0 | 0.00 | 3.5 | 1.6 | 0.00 | West Virginia<br>Beef 12.0 Dairy 3.9<br>Fruits... |
| **48** | WI | Wisconsin | state | 3090.23 | 107.3 | 38.6 | 34.5 | 633.60 | 42.8 | 91.0 | 133.80 | 56.8 | 92.2 | 148.99 | 460.5 | 96.7 | 0.00 | Wisconsin<br>Beef 107.3 Dairy 633.6<br>Fruits ... |
| **49** | WY | Wyoming | state | 349.69 | 75.1 | 33.2 | 0.1 | 2.89 | 0.1 | 0.1 | 0.17 | 3.9 | 6.3 | 10.23 | 9.0 | 20.7 | 0.00 | Wyoming<br>Beef 75.1 Dairy 2.89<br>Fruits 0.17... |

```
data = dict(type='choropleth',
            colorscale = 'Greens',
            locations = df['code'],
            z = df['total exports'],
            locationmode = 'USA-states',
            text = df['text'],
            marker = dict(line = dict(color = 'rgb(255,255,255)',width = 2)),
            colorbar = {'title':"Millions USD"}
            )

layout = dict(title = '2011 US Agriculture Exports by State',
              geo = dict(scope='usa',
                         showlakes = True,
                         lakecolor = 'rgb(85,173,240)')
             )

configure_plotly_browser_state()
choromap = go.Figure(data = [data],layout = layout)
iplot(choromap)
```




```
df=pd.read_csv('/content/drive/MyDrive/Wharton Acads/Python/2014_World_GDP')
```

```
df.head()
```

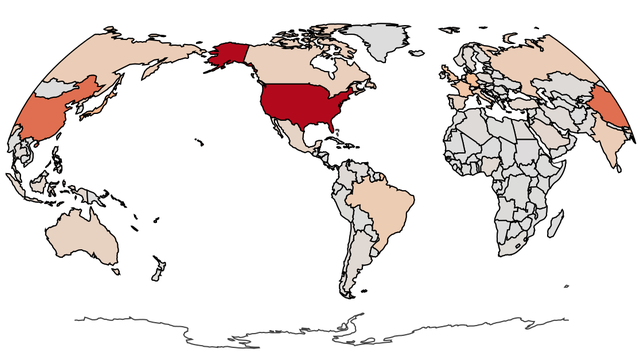
| | COUNTRY | GDP (BILLIONS) | CODE |
|---|---|---|---|
| **0** | Afghanistan | 21.71 | AFG |
| **1** | Albania | 13.40 | ALB |
| **2** | Algeria | 227.80 | DZA |
| **3** | American Samoa | 0.75 | ASM |

```
data = dict(
        type = 'choropleth',
        locations = df['CODE'],
        z = df['GDP (BILLIONS)'],
        text = df['COUNTRY'],
        colorbar = {'title' : 'GDP Billions US'},
      )
```

```
layout = dict(
    title = '2014 Global GDP',
    geo = dict(
        showframe = False,
        projection = {'type':'natural earth'}
    )
)
```

```
configure_plotly_browser_state()
choromap = go.Figure(data = [data],layout = layout)
iplot(choromap)
```

[object Object]



[object Object]