



GETTING STARTED

1. Install

In the terminal
`sudo pip install plotly`

2. Sign Up & Configure

<http://www.plot.ly/python/getting-started>

3. Boilerplate Imports

```
import plotly.plotly as py
import plotly.graph_objs as go

If offline, replace first line by:
import plotly.offline as py
```

4. A Hello World Figure

```
trace = {'x': [1, 2], 'y': [1, 2]}
data = [trace]
data = {}
fig = go.Figure (
    data = data, layout = layout )
```

5. Plot the Figure!

In the terminal:
`plot_url = py.plot (fig)`

Or in the IPython notebook:
`py.iplot (fig)`

BASIC CHARTS

Line Plots

```
trace1 = go.Scatter (
    x = [ 1, 2 ], y = [ 1, 2 ])
trace2 = go.Scatter (
    x = [ 1, 2 ], y = [ 2, 1 ])
py.iplot ( [ trace1, trace2 ])
```

Bubble Charts

```
trace = go.Scatter (
    x = [ 1, 2, 3 ], y = [ 1, 2, 3 ],
    marker = dict (
        color = [ 'red', 'blue',
        'green' ]
        size = [ 30, 80, 200 ] ),
    mode = 'markers' )
py.iplot ( [ trace ])
```

Scatter Plots

```
trace1 = go.Scatter (
    x = [ 1, 2, 3 ], y = [ 1, 2, 3 ],
    text = [ 'A','B','C' ],
    textposition = 'top center'
    mode = 'markers+text' )
mode = [ trace ]
py.iplot ( data )
```

Heatmaps

```
trace = go.Heatmap (
    z = [ [ 1, 2, 3, 4 ],
    [ 5, 6, 7, 8 ] ] )
data = [ trace ]
py.iplot ( data )
```

Bar Charts

```
trace = go.Bar (
    x = [ 1, 2 ], y = [ 1, 2 ])
data = [ trace ]
py.iplot ( data )
```

Area Plots

```
trace = go.Scatter (
    x = [ 1, 2 ], y = [ 1, 2 ],
    fill = 'tonexty' )
data = [ trace ]
py.iplot ( data )
```

LAYOUT

Legends

```
trace1 = go.Scatter (
    name = 'Calvin'
    x = [ 1, 2 ], y = [ 1, 2 ] )
```

```
trace2 = go.Scatter (
    name = 'Hobbes'
    x = [ 2, 1 ], y = [ 2, 1 ] )
```

```
layout = go.Layout (
    showlegend = True ,
    legend = dict (
        x = 0.2 , y = 0.5 )
    )
```

```
data = [ trace1 , trace2 ]
fig = go.Figure (
    data = data ,
    layout = layout )
py.iplot ( fig )
```

Axes

```
trace = go.Scatter (
    x = [ 1, 2, 3, 4 ],
    y = [ 1, 2, 3, 6 ] )
```

```
axis_template = dict (
    showgrid = False ,
    zeroline = False ,
    nticks = 20 ,
    showline = True ,
    title = 'X AXIS'
    mirror = 'all' )
layout = go.Layout (
    xaxis = axis_template ,
    yaxis = axis_template ,
    )
```

```
data = [ trace ]
fig = go.Figure (
    data = data
    layout = layout )
py.iplot ( fig )
```

STATISTICAL CHARTS

Histograms

```
trace = go.Histogram (
  x = [ 1, 2, 3, 3, 3, 4, 5 ] )
data = [ trace ]
py.iplot ( data )
```

Box Plots

```
trace = go.Box (
  x = [ 1, 2, 3, 3, 3, 4, 5 ] )
data = [ trace ]
py.iplot ( data )
```

2D Histogram

```
trace = go.Histogram2d (
  x = [ 1, 2, 3, 3, 3, 4, 5 ],
  y = [ 1, 2, 3, 3, 3, 4, 5 ] )
data = [ trace ]
py.iplot ( data )
```

MAPS

Bubble Map

```
trace = dict (
  type = 'scattergeo' ,
  lon = [ 100, 400 ], lat = [ 0, 0 ],
  marker = dict (
    marker = [ 'red', 'blue' ]
    size = [ 30, 50 ] ) ,
  mode = 'markers' )
py.iplot ( [ trace ] )
```

Choropleth Map

```
trc = dict (
  type = 'choropleth' ,
  locations = [ 'AZ', 'CA', 'VT' ],
  locationmode = 'USA-states' ,
  colorscale = [ 'Viridis' ],
  z = [ 10, 20, 40 ] )
lyt = dict ( geo = dict ( scope = 'usa' ) )
map = go.Figure ( data = [ trc ],
  layout = lyt )
py.iplot ( map )
```

Scatter Map

```
trace = dict (
  type = 'scattergeo' ,
  lon = [ 42, 39 ], lat = [ 12, 22 ],
  marker = [ 'Rome', 'Greece' ],
  mode = 'markers' )
py.iplot ( [ trace ] )
```

3D CHARTS

3D Surface Plots

```
trace = go.Surface (
  colorscale = 'Viridis' ,
  z = [ [ 3, 5, 8, 13 ],
    [ 21, 13, 8, 5 ] ] )
data = [ trace ]
py.iplot ( data )
```

3D Line Plots

```
trace = go.Scatter3D (
  x = [ 9, 8, 5, 1 ], y = [ 1, 2, 4, 8 ],
  z = [ 11, 8, 15, 3 ],
  mode = 'lines' )
data = [ trace ]
py.iplot ( data )
```

3D Scatter Plots

```
trace = go.Scatter3D (
  x = [ 9, 8, 5, 1 ], y = [ 1, 2, 4, 8 ],
  z = [ 11, 8, 15, 3 ],
  mode = 'markers' )
data = [ trace ]
py.iplot ( data )
```

FIGURE HIERARCHY

Figure { }

```
DATA [ ]
  TRACE { }
    x, y, z [ ]
    color, text, size [ ]
    colorscale ABC or [ ]
  MARKER { }
    color ABC
    symbol ABC
  LINE { }
    color ABC
    width 123

LAYOUT { }
  title ABC
  XAXIS, YAXIS { }
  SCENE { }
    XAXIS, YAXIS, ZAXIS { }
  GEO { }
  LEGEND { }
  ANNOTATIONS { }
```

{ } = dictionary
[] = list
ABC = string
123 = number