



GETTING STARTED

1. CDN

```
<head>
<script src =
'https://cdn.plot.ly/plotly-latest.
min.js'></script>
</head>
```

2. Sign Up & Configure

plot.ly/javascript/getting-started

3. A Hello World Figure

```
JS:
<script>
myDiv =
document.getElementById (
'myDiv' );
data = { x : [ 1, 2, 3, 4, 5 ],
y : [ 1, 2, 4, 8, 16 ] };
trace = [ data ];
Plotly.plot ( myDiv , trace );
</script>
```

```
HTML:
<head>
<script src =
'https://cdn.plot.ly/plotly-latest.
min.js'></script>
</head>
<div id = 'myDiv' ></div>
```

BASIC CHARTS

Line Plots

```
trace1 = {
x : [ 1, 2 ], y : [ 1, 2 ]
type = 'scatter' };
trace2 = {
x : [ 3, 4 ], y : [ 9, 16 ]
type = 'scatter' };
Plotly.plot (
div , [ trace1 , trace2 ] );
```

Scatter Plots

```
trace = {
x : [ 1, 2, 3 ],
y : [ 1, 2, 3 ]
text : [ 'A', 'B', 'C' ]
textposition : 'top center'
mode : 'markers+text' };
Plotly.plot ( div , [ trace ] );
```

Bar Charts

```
trace = {
x : [ 1, 2 ],
y : [ 1, 2 ]
type : 'bar' };
data = [ trace ];
Plotly.plot ( div , data );
```

Bubble Charts

```
trace {
x : [ 1, 2, 3 ],
y : [ 1, 2, 3 ],
marker : {
color : [ 'red', 'blue' ],
size : [ 20, 50, 80 ] },
mode : 'markers' };
Plotly.plot (
'myDiv', [ trace ] );
```

Heatmaps

```
trace = {
z : [[ 1, 2 ], [ 1, 2 ] ],
type : 'heatmap' },
data : [ trace ],
Plotly.plot (
'myDiv', data );
```

Area Plots

```
trace {
x : [ 1, 2, 3 ],
y : [ 1, 2, 3 ],
type : 'scatter',
type : 'tonexty' };
Plotly.plot (
'myDiv', [ trace ] );
```

LAYOUT

Legends

```
trace1 = {
x : [ 1, 2, 3 ],
y : [ 1, 2, 3 ]
name = 'Calvin',
type = 'scatter' };

trace2 = {
x : [ 1, 2, 3 ],
y : [ 1, 2, 3 ]
name = 'Hobbes',
type = 'scatter' };

layout = {
showlegend = true ,
legend : {
x : 0.2,
y : 0.5 } };

fig = {
data : [ trace1 , trace2 ],
layout : layout };
Plotly.plot (
'myDiv', fig );
```

Axes

```
trace = {
x : [ 1, 2, 3 ],
y : [ 1, 2, 3 ]
type = 'scatter' };

axis_template = {
showgrid = false ,
zeroline = false ,
nticks = 20 ,
showline = true ,
title = 'X Axis'
mirror = 'all' }

layout = {
xaxis = axis_template ,
yaxis = axis_template };

fig = {
data : [ trace ],
layout : layout };
Plotly.plot (
'myDiv', fig );
```

STATISTICAL CHARTS

Histograms

```
var trace = {  
  x: [ 1, 2, 3, 4, 5 ],  
  type = 'histogram'  
Plotly.Plot (  
  'Div', [ trace ] );
```

Box Plots

```
var trace = {  
  y: [ 1, 2, 3, 4, 5 ],  
  type = 'box'  
Plotly.Plot (  
  'Div', [ trace ] );
```

2D Histogram

```
var trace = {  
  x: [ 1, 2, 3, 4, 5 ],  
  y: [ 1, 2, 3, 4, 5 ],  
  type = 'histogram2d'  
Plotly.Plot (  
  'Div', [ trace ] );
```

MAPS

Bubble Map

```
trace = {  
  type: 'scattergeo',  
  lon: [ 100, 400 ],  
  lat: [ 0, 0 ],  
  marker: { color: [ 'red', 'blue' ],  
  size: [ 30, 50 ],  
  mode = 'markers'  
Plotly.Plot (  
  'myDiv', [ trace ] );
```

Choropleth Map

```
trace = {  
  type = 'scattergeo',  
  locations = [ 'AZ', 'CA', 'VT' ],  
  locationmode = 'USA-states',  
  colorscale = 'Viridis',  
  z = [ 10, 20, 40 ] };  
layout = { geo: { scope = 'usa' } };  
fig = { data: [ trace ], layout: layout };  
Plotly.newPlot (  
  'myDiv', fig );
```

Scatter Map

```
trace = {  
  type = 'scattergeo',  
  lon = [ 42, 39 ],  
  lat = [ 12, 22 ],  
  text = [ 'Rome', 'Greece' ],  
  mode = 'markers';  
Plotly.newPlot (  
  'myDiv', [ trace ] );
```

3D CHARTS

3D Surface Plots

```
trace = {  
  colorscale = 'Viridis',  
  z = [ [ 3, 5, 7, 9 ],  
  [ [ 21, 13, 8, 5 ] ] ];  
Plotly.newPlot (  
  'myDiv', [ trace ] );
```

3D Line Plots

```
trace = {  
  x = [ 9, 8, 5, 1 ],  
  y = [ 1, 2, 4, 8 ],  
  z = [ 11, 8, 15, 3 ],  
  mode = 'lines';  
data = [ trace ];  
Plotly.newPlot (  
  'myDiv', data );
```

3D Scatter Plots

```
trace = {  
  x = [ 9, 8, 5, 1 ],  
  y = [ 1, 2, 4, 8 ],  
  z = [ 11, 8, 15, 3 ],  
  mode = 'markers';  
Plotly.newPlot (  
  'myDiv', [ trace ] );
```

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  
  showlegend True/False  
  autosize True/False  
  XAXIS, YAXIS {}  
  SCENE {}  
    XAXIS, YAXIS, ZAXIS {}  
  GEO {}  
  LEGEND {}  
  ANNOTATIONS {}
```

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