Date 07-07-2021

Exercise Create a Simple Dashboard

In [1]:
```
import plotly.offline as pyo
import plotly.graph_objs as go
from plotly import subplots
import plotly
import dash
import dash_core_components as dcc
import dash_html_components as html
import pandas as pd

versions_of_modules_used = {dash.__name__ : dash.__version__,
                            dcc.__name__ : dcc.__version__,
                            plotly.__name__ : plotly.__version__,
                            html.__name__ : html.__version__,
                            pd.__name__ : pd.__version__}

for i, j in versions_of_modules_used.items():
        print(i,"=",j)
```

```
dash = 1.20.0
dash_core_components = 1.16.0
plotly = 5.1.0
dash_html_components = 1.1.3
pandas = 1.1.3
```

Module Name	Module Versions
dash	1.20.0
dash_core_components	1.16.0
plotly	5.1.0
dash_html_components	1.1.3
pandas	1.1.3

In [2]:
```
old_faithful_data_csv = pd.read_csv("OldFaithful.csv", usecols = ['Y', 'X'])
old_faithful_data_csv
```

Out[2]:

	Y	X
0	78	4.4
1	74	3.9
2	68	4.0
3	76	4.0
4	80	3.5
...	...	...
217	61	2.1
218	81	4.2
219	48	2.1
220	84	5.2
221	63	2.0

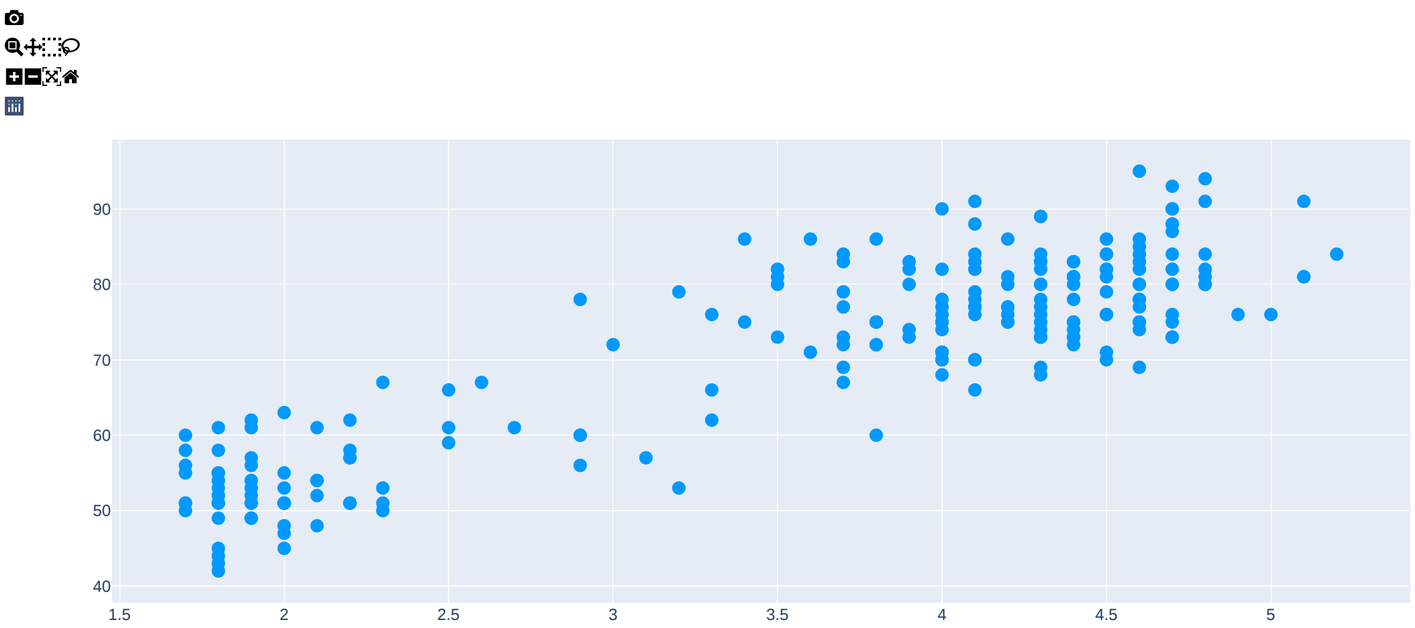
222 rows × 2 columns

In [3]:
```
data = [go.Scatter(x = old_faithful_data_csv['X'],
                   y = old_faithful_data_csv['Y'],
                   mode = 'markers',
                   hovertemplate='Duration of eruption (minutes): %{x}<br>Interval to next eruption (minutes): %{y}',
                   name='A Dash<br>App<br>Exercise',
                   marker = dict(size = 10,
                                 color = "#0099ff"))]
```

In [4]:
```
layout = go.Layout(title = 'Old Faithful Eruption Intervals v Durations',
                   xaxis = {'title': 'Duration of eruption (minutes)'},
                   yaxis = {'title': 'Interval to next eruption (minutes)'},
                   hovermode='closest',
                   title_x = 0.5)
```

In [5]:
```
fig = go.Figure(data, layout)
```

In [6]:
```
fig.show()
```



In [7]:
```
pyo.plot(data, filename="tutorial_1 (Exercise Create a Simple Dashboard){Graph}.html")
```

Out[7]:
```
'tutorial_1 (Exercise Create a Simple Dashboard){Graph}.html'
```

In [8]:
```
heading_style = {'font-size' : '50px',
'line-height' : '40px',
'margin' : '1em 0 .6em 0',
'font-weight' : 'normal',
'color' : 'white',
'font-family' : 'Hammersmith One',
'text-shadow' : '0 1px 0 rgba(0,0,0,0.4)',
'position' : 'relative',
'color' : '#6Cf',
'text-align' : 'center',}
```

In [9]:
```
app = dash.Dash()
app.layout = html.Div(children = [html.H1("Exercise Create a Simple Dashboard", style = heading_style),
                                  dcc.Graph(id = "ScatterPlots for Exercise",
                                            figure = fig)])
app.run_server()
```

```
Dash is running on http://127.0.0.1:8050/

 * Serving Flask app "__main__" (lazy loading)
 * Environment: production
   WARNING: This is a development server. Do not use it in a production deployment.
   Use a production WSGI server instead.
 * Debug mode: off
 * Running on http://127.0.0.1:8050/ (Press CTRL+C to quit)
127.0.0.1 - - [07/Jul/2021 17:37:23] "GET / HTTP/1.1" 200 -
127.0.0.1 - - [07/Jul/2021 17:37:23] "GET /_dash-layout HTTP/1.1" 200 -
127.0.0.1 - - [07/Jul/2021 17:37:23] "GET /_dash-dependencies HTTP/1.1" 200 -
```