



## IBM Developer SKILLS NETWORK

# Dash Components

## Objectives

- Create a dash application layout
- Add HTML H1, P, and Div components
- Add core graph component
- Add multiple charts

## Dataset Used

[Airline Reporting Carrier On-Time Performance](#) dataset from [Data Asset eXchange](#)

## Lab Questions

We will be using the same pie and sunburst chart theme from Plotly basics lab.

### *Theme for Pie Chart*

Proportion of distance group (250 mile distance interval group) by month (month indicated by numbers).

### *Theme for Sunburst Chart*

Hierarchical view in othe order of month and destination state holding value of number of flights.

```
In [1]: # Import required packages
import pandas as pd
```

```
import plotly.express as px
```

## Load the data

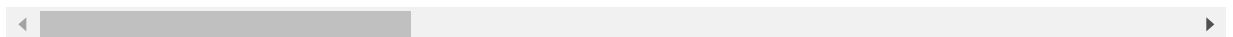
```
In [2]: # Read the airline data into pandas dataframe
# Read the airline data into pandas dataframe
airline_data = pd.read_csv('https://cf-courses-data.s3.us.cloud-object-storage.appd
                        encoding = "ISO-8859-1",
                        dtype={'Div1Airport': str, 'Div1TailNum': str,
                              'Div2Airport': str, 'Div2TailNum': str})
```

```
In [3]: # Preview the first 5 lines of the loaded data
airline_data.head()
```

```
Out[3]:
```

	Unnamed: 0	Year	Quarter	Month	DayofMonth	DayOfWeek	FlightDate	Reporting_Airline	DOT_
0	1295781	1998	2	4	2	4	1998-04-02	AS	
1	1125375	2013	2	5	13	1	2013-05-13	EV	
2	118824	1993	3	9	25	6	1993-09-25	UA	
3	634825	1994	4	11	12	6	1994-11-12	HP	
4	1888125	2017	3	8	17	4	2017-08-17	UA	

5 rows × 110 columns



```
In [4]: # Shape of the data
airline_data.shape
```

```
Out[4]: (27000, 110)
```

```
In [5]: # Randomly sample 500 data points. Setting the random state to be 42 so that we get
data = airline_data.sample(n=500, random_state=42)
```

```
In [6]: # Get the shape of the trimmed data
data.shape
```

```
Out[6]: (500, 110)
```

Proportion of distance group (250 mile distance interval group) by month  
(month indicated by numbers).

```
In [7]: # Pie Chart Creation  
fig = px.pie(data, values='Month', names='DistanceGroup', title='Distance group prop  
fig.show()
```

## Let's start creating dash application

### Theme

Proportion of distance group (250 mile distance interval group) by month (month indicated by numbers).

### To do:

1. Import required libraries and create an application layout
2. Add title to the dashboard using HTML H1 component
3. Add a paragraph about the chart using HTML P component
4. Add the pie chart created above using core graph component
5. Run the app

### Hints

General examples can be found [here](#).

- For step 1 (only review), this is very specific to running app from Jupyterlab.
  - For Jupyterlab, we will be using `jupyter-dash` library. Adding `from jupyter_dash import JupyterDash` import statement.
  - Instead of creating dash application using `app = dash.Dash()`, we will be using `app = JupyterDash(__name__)`.
- For step 2,
  - [Plotly H1 HTML Component](#)
  - Title as `Airline Performance Dashboard`
  - Use `style` parameter and make the title center aligned, with color code `#503D36`, and font-size as 40. Check `More about HTML` section [here](#).
- For step 3,
  - [Plotly Paragraph Component](#)
  - Paragraph as `Proportion of distance group (250 mile distance interval group) by month (month indicated by numbers)`.
  - Use `style` parameter to make the description center aligned and with color `#F57241`.
- For step 4, refer [dcc.Graph](#) component usage.
- For step 5, you can refer examples provided [here](#).

NOTE: Run the solution cell multiple times if you are not seeing the result.

## App Skeleton

```
import dash

from jupyter_dash import JupyterDash

app = JupyterDash(__name__)
JupyterDash.infer_jupyter_proxy_config()

app.layout = html.Div(children=[html.H1(.....),
                                html.P(.....),
                                dcc.Graph(.....)
                                ])

if __name__ == '__main__':
    app.run_server(mode="inline", host="localhost")
```

```
In [14]: # Import required libraries
import dash
import dash_html_components as html
import dash_core_components as dcc
from jupyter_dash import JupyterDash
```

```
In [16]: JupyterDash.infer_jupyter_proxy_config()
```

```
In [18]: # needs to be run again in a separate cell due to a jupyterdash bug
JupyterDash.infer_jupyter_proxy_config()
```

```
In [19]: # Create a dash application
app = JupyterDash(__name__)

# Get the layout of the application and adjust it.
# Create an outer division using html.Div and add title to the dashboard using html.
# Add description about the graph using HTML P (paragraph) component
# Finally, add graph component.
app.layout = html.Div(children=[html.H1('Airline Dashboard',
                                         style={'textAlign': 'center',
                                               'color': '#503D36',
                                               'font-size': 40}),
                                html.P('Proportion of distance group (250 mile dista
                                         style={'textAlign': 'center', 'color': '#F572
                                dcc.Graph(figure=fig)])

if __name__ == '__main__':
    app.run_server(mode="inline", host="localhost")
```

/home/jupyterlab/conda/envs/python/lib/python3.6/site-packages/jupyter\_dash/jupyter\_

app.py:139: UserWarning:

The 'environ['werkzeug.server.shutdown']' function is deprecated and will be removed in Werkzeug 2.1.

Double-click **here** for the solution.

## Summary

Congratulations for completing your dash basics lab.

In this lab, you have learnt how to use dash HTML and core components for creating dashboard.

## Author

[Saishruthi Swaminathan](#)

## Changelog

Date	Version	Changed by	Change Description
12-18-2020	1.0	Nayef	Added dataset link and upload to Git

© IBM Corporation 2020. All rights reserved.

In [ ]: