

# Building Web Map with Python and Folium

## Python

- Python is an interpreted, object-oriented, high-level programming language with dynamic semantics. Its high-level built in data structures, combined with dynamic typing and dynamic binding, make it very attractive for Rapid Application Development, as well as for use as a scripting or glue language to connect existing components together. Python's simple, easy to learn syntax emphasizes readability and therefore reduces the cost of program maintenance. Python supports modules and packages, which encourages program modularity and code reuse. The Python interpreter and the extensive standard library are available in source or binary form without charge for all major platforms, and can be freely distributed.

## Folium

Folium is a powerful Python library that helps you create several types of Leaflet maps. The fact that the Folium results are interactive makes this library very useful for dashboard building. To get an idea, just zoom/click around on the next map to get an impression

## Pandas DataFrames

Pandas is a high-level data manipulation tool developed by Wes McKinney. It is built on the Numpy package and its key data structure is called the DataFrame. DataFrames allow you to store and manipulate tabular data in rows of observations and columns of variables.

We will load the data from [The volcanoes point locations](#)

We can use the folium and pandas library by installing them through the command prompt

```
pip install pandas
pip install folium
```

```
map = folium.Map(location=[38.58, -99.09], zoom_start=6, tiles="Mapbox Bright")

fgv = folium.FeatureGroup(name="Volcanoes")

for lt, ln, el in zip(lat, lon, elev):
    fgv.add_child(folium.CircleMarker(location=[lt, ln], radius = 6, popup=str(el)+" m",
    fill_color=color_producer(el), fill=True, color = 'grey', fill_opacity=0.7))

fgp = folium.FeatureGroup(name="Population")

fgp.add_child(folium.GeoJson(data=open('world.json', 'r', encoding='utf-8-sig').read(),
style_function=lambda x: {'fillColor':'green' if x['properties']['POP2005'] < 10000000
else 'orange' if 10000000 <= x['properties']['POP2005'] < 20000000 else 'red'}))
```

Above functions creates the base map aka ground layer of the map where we can see the names of the folders, also we have assigned some of the point markers

We can also use the colourMarker or just 'marker' in the folium library, In the above code we have used the CircleMarker.

So our map contains two layers –base layer and custom layer which has circle markers, may also be called as point layer.

```
fgp.add_child(folium.GeoJson(data=open('world.json', 'r', encoding='utf-8-sig').read(),
style_function=lambda x: {'fillColor':'green' if x['properties']['POP2005'] < 10000000
else 'orange' if 10000000 <= x['properties']['POP2005'] < 20000000 else 'red'}))
```

This function adds another layer , called as Polygon layer which represents the areas as well as world population.

So countries are represented by the areas which we can differentiate them with colours , hence we will use some data [The GeoJson world polygon layer with population data](#) and we get the results

## Snapshot

