220915_folium

September 17, 2022

1 Folium

folium makes it easy to visualize data that's been manipulated in Python on an interactive leaflet map.

1.0.1

, .

- (, : latitude)
- : (: , : , : longitude) . (°) , 180°E (180) 180° W (180) .
- (, : prime meridian) , . 0° 1884

[1]:
```
# pip install folium
# pip install folium --use-feature=2020-resolver

import folium
print(folium.__version__)
```

```
0.12.1.post1
```

[2]:
```
#  ,
loc = folium.Map(location=[35.162441, 126.910339]) # gwangju
loc
```

[2]:
```
<folium.folium.Map at 0x19e72fed9d0>
```

[3]:
```
# London
loc = folium.Map(location=[51.507351, -0.127758])
loc
```

[3]:
```
<folium.folium.Map at 0x19e72ff0f70>
```

1.0.2

[4]:
```
#    ,    (zoom_start        .)
# default = 10, zoom_start . 0~18
m = folium.Map(
    location=[35.162441, 126.910339],
    zoom_start= 15
)

m
```

[4]:
```
<folium.folium.Map at 0x19e72ff4eb0>
```

1.0.3 html

[5]:
```
m.save('Gwangju_map.html')
```

[6]:
```
#
import os
path = os.getcwd()
os.listdir(path)
```

[6]:
```
['.ipynb_checkpoints',
 '220914_matplotlib2.ipynb',
 '220914_seaborn.ipynb',
 '220915_folium.ipynb',
 'data',
 'Gwangju_map.html',
 'map_circle.html',
 'Seoul_map.html']
```

1.0.4 (Marker)

[7]:
```
#
loc = folium.Map(location=[35.162441, 126.910339], zoom_start=12)

folium.Marker([35.17690, 126.90690]).add_to(loc) #
folium.Marker([35.20743, 126.86369]).add_to(loc) #
loc
```

[7]:
```
<folium.folium.Map at 0x19e730777f0>
```

1.0.5 popup icon

- Color

['red', 'blue', 'green', 'purple', 'orange', 'darkred', 'lightred', 'beige', 'darkblue', 'darkgreen', 'cadetblue', 'darkpurple', 'white', 'pink', 'lightblue', 'lightgreen', 'gray', 'black', 'lightgray']

[8]:
```
m = folium.Map(
    location=[35.162441, 126.910339],
    zoom_start=13
)

folium.Marker(
    location=[35.17690, 126.90690],
    popup = 'Chonnam National University',                          #     popup
    icon = folium.Icon(color='purple', icon="info-sign") #     #␣
 ↪glyphicon-check, star
).add_to(m)

m
```

[8]:
```
<folium.folium.Map at 0x19e73082610>
```

1.0.6 CircleMarker

[9]:
```
m = folium.Map(
    location=[35.17690, 126.90690],
    zoom_start=13
)

folium.CircleMarker(
    location=[35.17690, 126.90690],
    radius = 100,
    color = '#ffffgg',
    fill_color='#fffggg',
    popup = 'yanghwajin',                                #     popup
).add_to(m)

m
```

[9]:
```
<folium.folium.Map at 0x19e730908b0>
```

1.0.7 Marker - MarkerCluster

Marker - MarkerCluster Adds a MarkerCluster layer on the map.

[10]:
```
from folium import plugins
import numpy as np
import os
```

[11]:
```
N = 100

data = np.array(
    [
```

```
     np.random.uniform(low=35, high=60, size=N),
     np.random.uniform(low=-12, high=30, size=N),
    ]
).T

print(data[0:10])
```

```
[[ 47.4668133  -10.96650545]
 [ 37.39324164   6.82227588]
 [ 43.47063167  -4.38889782]
 [ 40.86870636  20.51436967]
 [ 52.0846408    8.44637679]
 [ 58.51753043  29.40899666]
 [ 56.56225478  -4.34889999]
 [ 40.75290813   5.78486667]
 [ 53.34720017  14.80977098]
 [ 37.52480206  25.50454478]]
```

1.0.8 for Circle

[12]:
```
print(list(data[0]))
print(list(data[1]))
print(list(data[2]))
print(list(data[3]))
```

```
[47.466813297360375, -10.966505449246341]
[37.39324163563483, 6.822275884598707]
[43.47063166702563, -4.388897815573145]
[40.86870636440581, 20.514369672332535]
```

[13]:
```
m = folium.Map(
    location = [35.36462578, 0.33840394],
    zoom_start = 6,
    tiles = 'Cartodb Positron'
)

for i in range(len(data)):
    folium.Circle(
        location = list(data[i]),
        radius = 50,
        color = '#000000',
        fill = 'crimson'
    ).add_to(m)

m.save('map_circle.html')
m
```

[13]:
```
<folium.folium.Map at 0x19e7306ff70>
```

[14]:
```
popups = [str(i) for i in range(N)]  # Popups texts are simple numbers.

m = folium.Map([45, 3], zoom_start=4)
plugins.MarkerCluster(data, popups=popups).add_to(m)
#m.save(os.path.join('.', 'Plugins_1.html'))

m
```

[14]:
```
<folium.folium.Map at 0x19e73149550>
```

1.0.9 Heatmap

.

[15]:
```
from folium.plugins import HeatMap

m = folium.Map(
    location = [35.36462578, 0.33840394],
    zoom_start = 5,
    tiles = 'Cartodb Positron'
)

HeatMap(data).add_to(m)
m
```

[15]:
```
<folium.folium.Map at 0x19e73220670>
```

1.0.10 GeoJson

[16]:
```
m = folium.Map(
    location=[37.5838699,127.0565831],
    zoom_start=10
)

m
```

[16]:
```
<folium.folium.Map at 0x19e7323f400>
```

seoul_municipalities_geo.json

[17]:
```
import json
with open('./data/seoul_municipalities_geo.json',mode='rt',encoding='utf-8') as␣
↪f:
    geo = json.loads(f.read())
    f.close()
```

```
folium.GeoJson(
    geo,
    name='seoul_municipalities'
).add_to(m)

# m.save('map.html')
m
```

[17]: <folium.folium.Map at 0x19e7323f400>

1.0.11 tiles

[18]:
```
stamen = folium.Map(location=[45.5236, -122.6750],
                    tiles='Stamen Toner',
                    zoom_start=13)
stamen
```

[18]: <folium.folium.Map at 0x19e73871c40>

[19]:
```
stamen = folium.Map(location=[45.5236, -122.6750],
                    tiles='Stamen Terrain',
                    zoom_start=13)
stamen
```

[19]: <folium.folium.Map at 0x19e73871460>

1.0.12 Ref

- https://python-visualization.github.io/folium/modules.html
- https://dailyheumsi.tistory.com/85
- https://raw.githubusercontent.com/southkorea/seoul-maps/master/kostat/2013/json/seoul_municipalities_