

# Folium 알아보기

## 학습 목표

- 위도, 경도 정보를 이용하여 지도 위에 정보를 표시하는 것을 실습을 통해 알아본다.

## 학습 내용

- 지도 시각화 라이브러리에 대해 알아본다.
- Folium을 이용하여 지도위에 위치를 표시하는 것을 알아본다.

### 01. 사전 이해

- 위도 (緯度, 영어: latitude )는 지구 상에서 적도 를 기준으로 북쪽 또는 남쪽으로 얼마나 떨어져 있는지 나타내는 위치
- 경도 : 경도 ( 문화어: 날줄, 한국 한자 : 經度, 영어: longitude )는 지구 상에서 본초 자오선 을 기준으로 동쪽 또는 서쪽으로 얼마나 떨어져 있는지 나타내는 위치이다. 경도의 단위는 도 (°)이며, 180°E (동경 180도) 부터 180° W (서경 180도) 까지의 범위 안에 있다.
- 본초 자오선 (本初子午線, 영어: prime meridian )은 영국 그리니치 의 그리니치 천문대 를 지나는 자오선 으 로, 경도 의 기준이 된다. 현재의 경도 0°로 1884년 에 국제 회의에서 그리니치 천문대를 지나는 본초 자오선 을 표준으로 삼기로 결정했다.

### 02. folium 설치 및 이슈

```
pip install folium
```

에러 발생 시,

```
ERROR: After October 2020 you may experience errors when installing or updating packages.  
This is because pip will change the way that it resolves dependency conflicts.
```

We recommend you use `--use-feature=2020-resolver` to test your packages with the new resolver before it becomes the default.

```
selenium 4.1.0 requires urllib3[secure]>=1.26, but you'll have urllib3 1.25.11 which is incompatible.
```

해결

```
pip install folium --use-feature=2020-resolver
```

In [1]:

```
# 위도(latitude), 경도(longitude) 를 이용한 위치표시
import folium

print(folium.__version__)
```

0.12.1.post1

In [2]:

```
# 위도, 경도 위치를 중심으로 한 지도를 표시
loc = folium.Map(location=[37.546817, 126.911411])
loc
```

Out[2]:



In [3]:

```
# 51.48715045410139, -0.00427069814187904  
loc = folium.Map(location=[51.48715, 0])  
loc
```

Out[3]:



## 지도 확대 및 축소

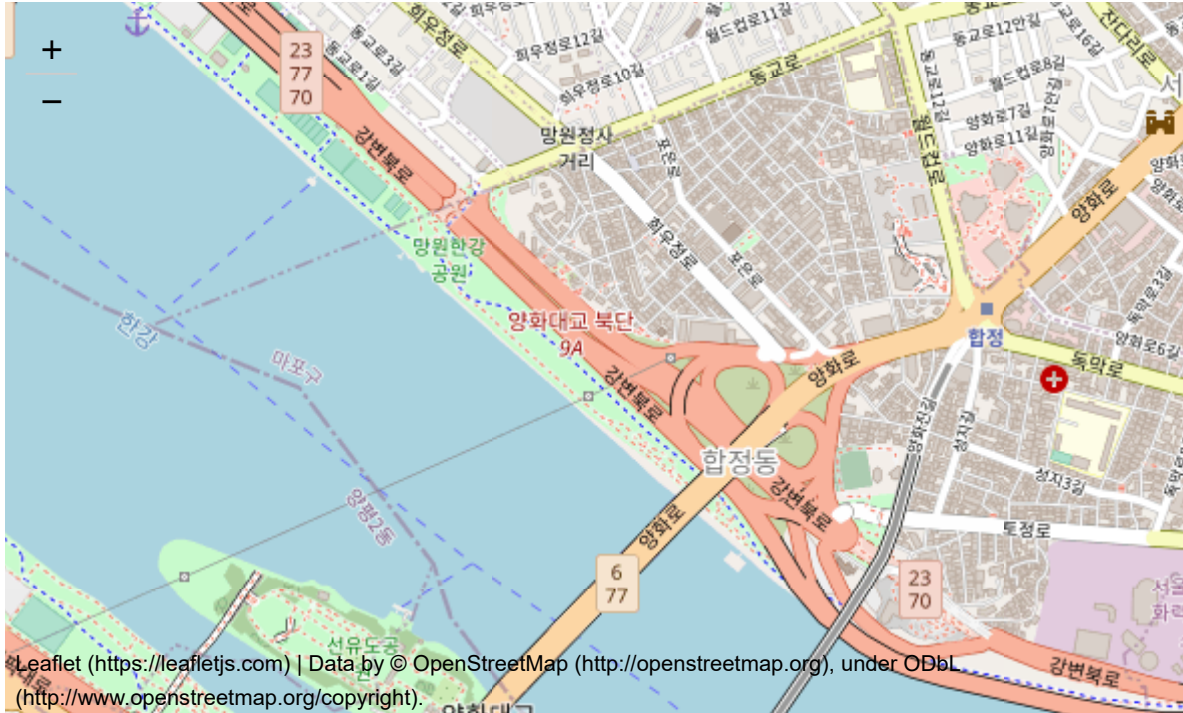
In [5]:

```
# 위치 표시, 확대 (zoom_start 값이 크면 확대됨.)
# default = 10, zoom_start . 0~18
import folium

m = folium.Map(
    location=[37.5466423, 126.9092281],
    zoom_start=15
)

m
```

Out[5]:



## html 파일로 저장

In [21]:

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

In [24]:

```
### 확인하기
import os
path = os.getcwd()
os.listdir(path)
```

Out[24]:

```
['.DS_Store',
 '.ipynb_checkpoints',
 'C1_2B_1_matplotlib_datavis_corona.ipynb',
 'C1_2_1_matplotlib.ipynb',
 'C1_3_1_Seaborn_Basic.ipynb',
 'C1_3_2_titanic_datavis.ipynb',
 'C1_3_3_Titanic_EDA_DataPreprocessing.ipynb',
 'C1_4_1_pandas_01.ipynb',
 'C1_4_2_pandas_02_california.ipynb',
 'C1_4_3_titanic_dataset.ipynb',
 'C1_4_4_화장품관련키워드분석_ing.ipynb',
 'C1_4_5_titanic_dataset_pandas_etc.ipynb',
 'C1_4_6_corona_analysis.ipynb',
 'C1_5_1_folium_local.ipynb',
 'C1_5_1_folium_withColab.ipynb',
 'C1_5_2_folium_seoul_data.ipynb',
 'C1_6_1_plotly_basic_iplot.ipynb',
 'C1_6_2_plotly_express_v11.ipynb',
 'data',
 'html_pdf',
 'map.html',
 'part2_1_1_datavis',
 'part2_1_2B_matplotlib',
 'part2_1_2_matplotlib',
 'part2_1_3_seaborn',
 'part2_1_4A_pandas',
 'part2_1_4B_pandas',
 'part2_1_5B_folium_project',
 'part2_1_5C_folium_실습결과물',
 'part2_1_5_folium',
 'part2_1_6B_plotly',
 'part2_1_6C_실습결과물',
 'part2_1_6_plotly',
 'part2_2_1A_konlpy',
 'part2_2_1B_konlpy_pratice',
 'part2_2_1C_konlpy_결과물',
 'Plugins_1.html']
```

## 지도의 위치 찍기(Marker)



In [25]:

```
## 원모양의 마커
## 양화진, 성산 중학교 마커
## folium.Marker, folium.CircleMaker
## 구글 지도 이용하여 위도, 경도 확인하기
loc = folium.Map(location=[37.546817, 126.911411], zoom_start=17)

folium.Marker([37.546817, 126.911411]).add_to(loc) # 양화진
folium.Marker([37.547394, 126.915816]).add_to(loc) # 성산중학교
loc
```

Out[25]:



## popup과 icon을 사용해보기

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

In [26]:

```
m = folium.Map(  
    location=[37.5466423,126.9092281],  
    zoom_start=15  
)  
  
folium.Marker(  
    location=[37.5466423,126.9092281],  
    popup = 'yanghwajin', # 선택 시 popup  
    icon = folium.Icon(color='purple', icon="info-sign") # 아이콘 # glyphicon-check, star  
) .add_to(m)  
  
m
```

Out[26]:



## CircleMarker 이용

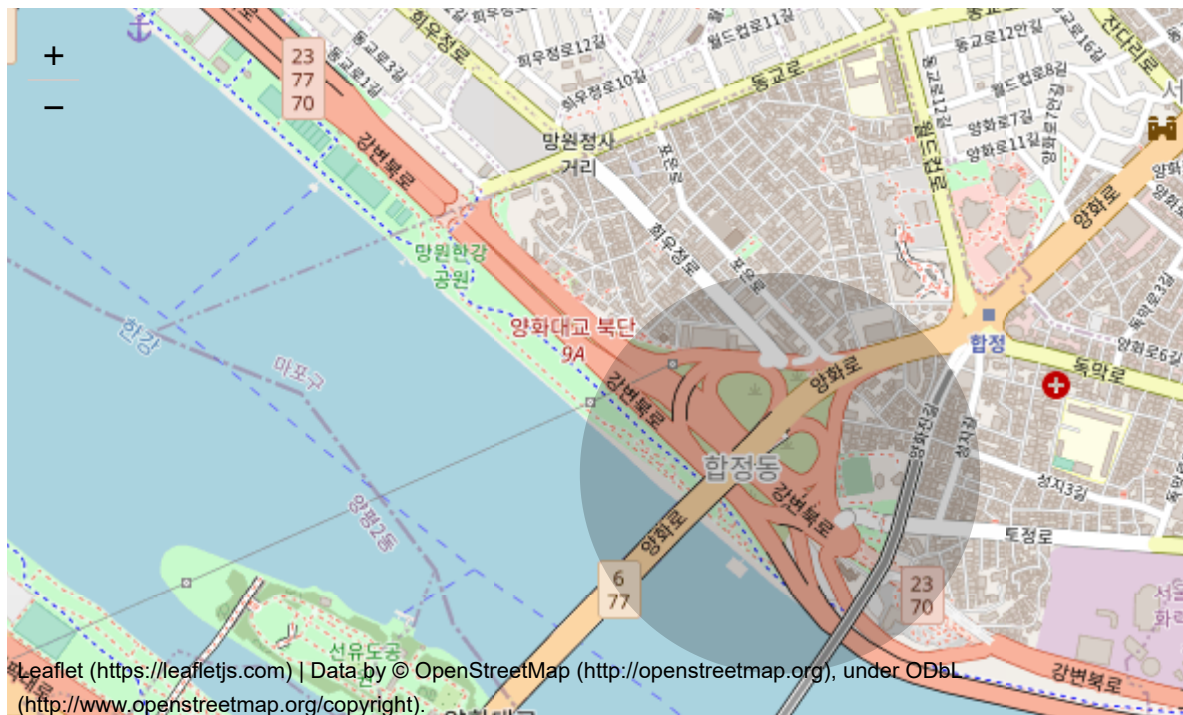
In [27]:

```
m = folium.Map(
    location=[37.5466423, 126.9092281],
    zoom_start=15
)

folium.CircleMarker(
    location=[37.5466423, 126.9092281],
    radius = 100,
    color = '#ffffgg',
    fill_color='#ffffgg',
    popup = 'yanghwajin',
    # 선택 시 popup
).add_to(m)

m
```

Out[27]:



In [28]:

```
from folium import plugins
```

## Marker를 군집화 시키기 - MarkerCluster

- Adds a MarkerCluster layer on the map.

In [29]:

```
import numpy as np
import os
```



In [30]:

```
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])
```

```
[[35.85406988  7.4665706 ]
 [35.36462578  0.33840394]
 [44.12642911 11.05151343]
 [45.47364162 -3.39706172]
 [50.23665313 19.44721392]
 [35.13422815  7.00823338]
 [40.04159914  4.18638371]
 [49.3120774  28.48541779]
 [49.0481822  29.04306931]
 [51.31981765 -0.97675987]]
```

## for문을 이용한 위치 Circle표시

In [54]:

```
print( list(data[1]) )
print( list(data[2]) )
print( list(data[3]) )
```

```
[35.364625784722605, 0.3384039352661681]
[44.1264291094885, 11.051513425207261]
[45.473641616893445, -3.3970617170088673]
```

In [57]:

```
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
```

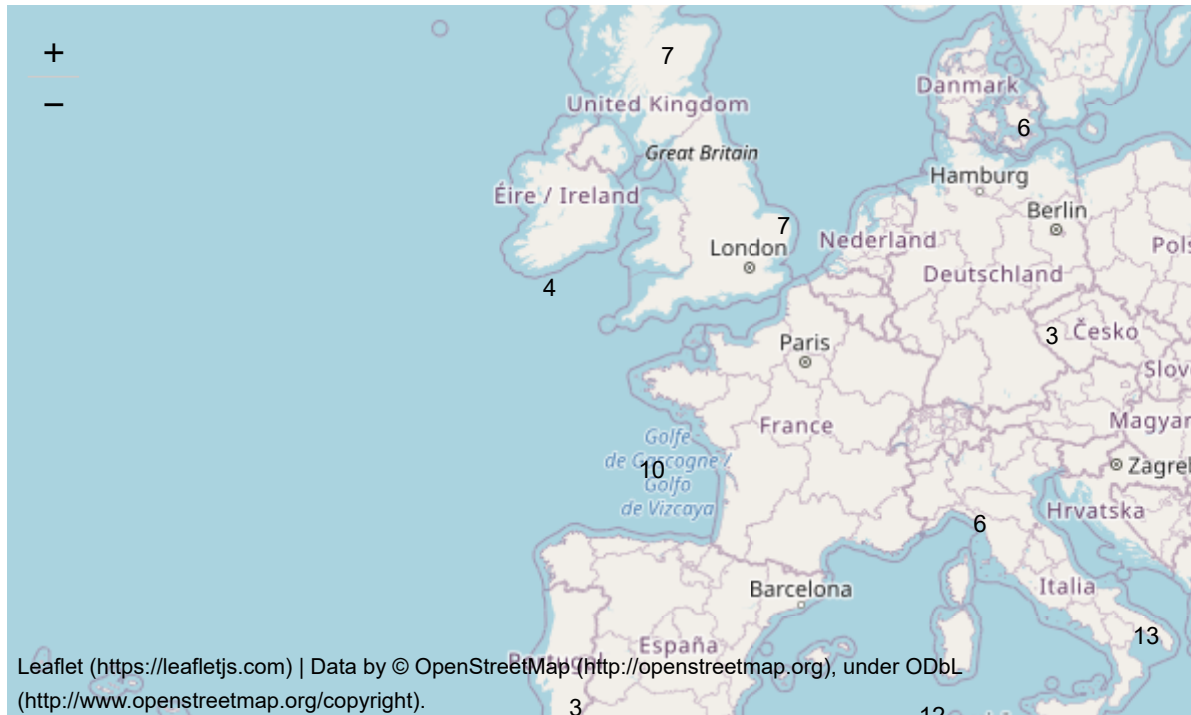
Out[57]:



In [42]:

```
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
```

Out[42]:



## Heatmap

- 어느 지역에 좌표들이 밀집되어 있는지를 잘 표현해 준다.

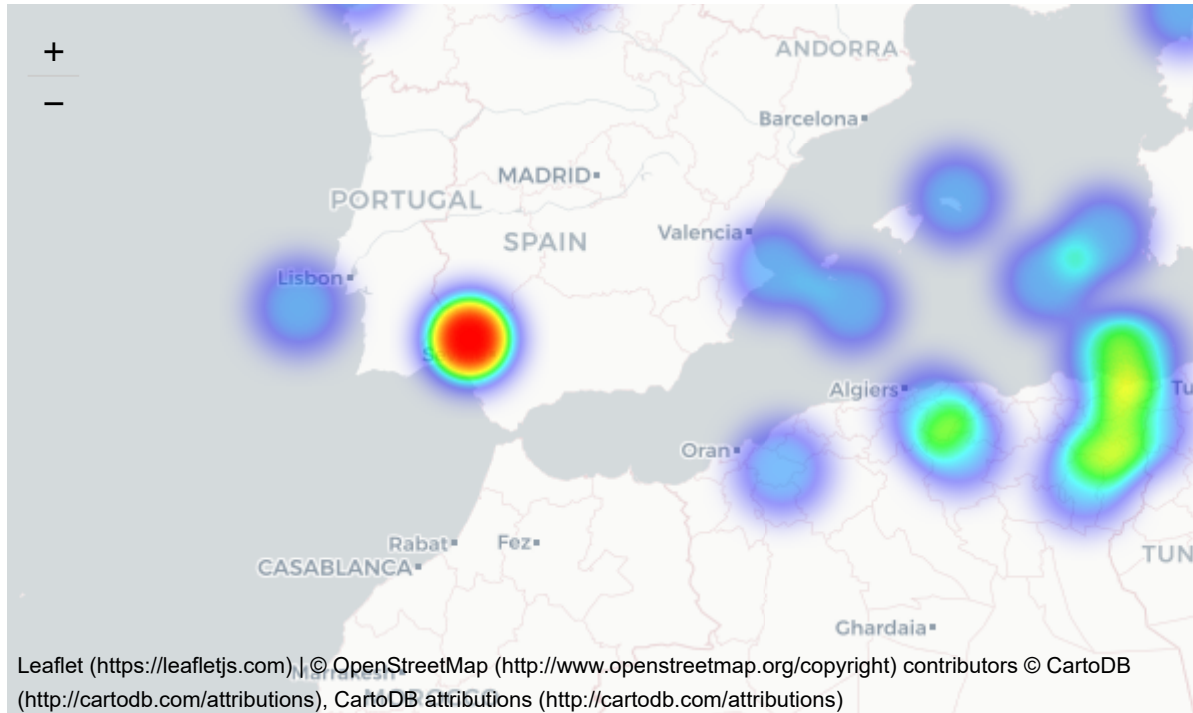
In [60]:

```
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
```

Out[60]:



## GeoJson 행정 구역 데이터

In [32]:

```
import folium

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

- seoul\_municipalities\_geo.json는 서울시 행정구역 데이터

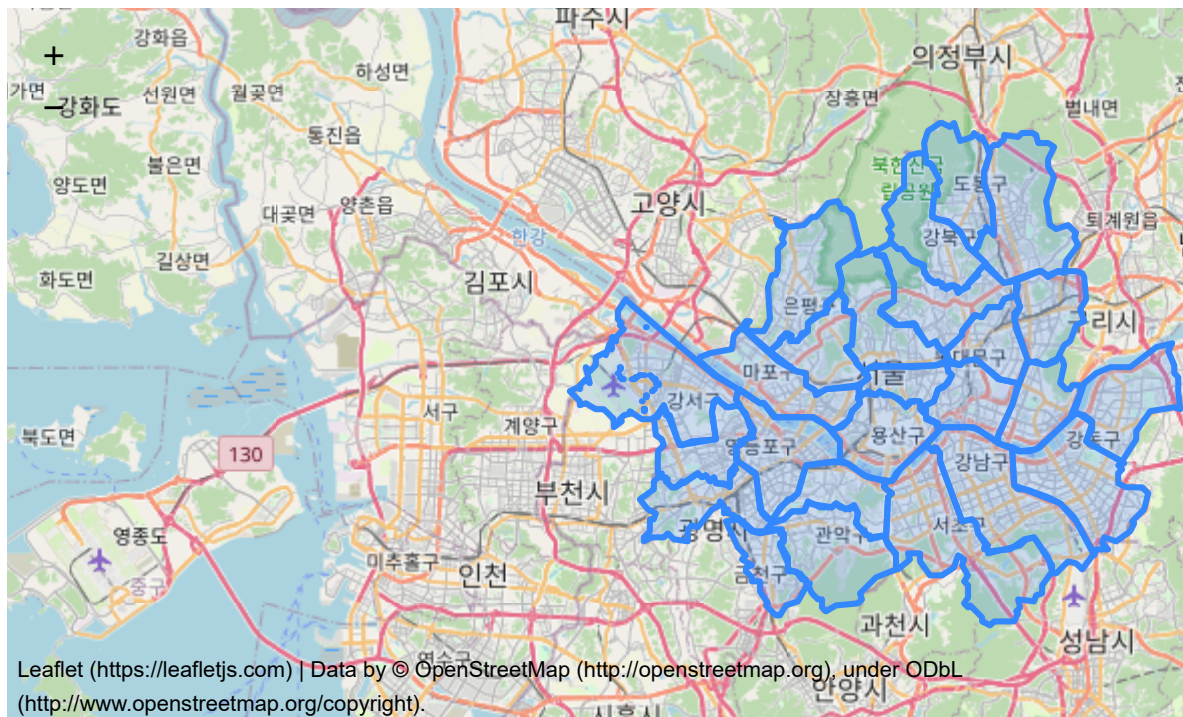
In [33]:

```
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
```

Out[33]:



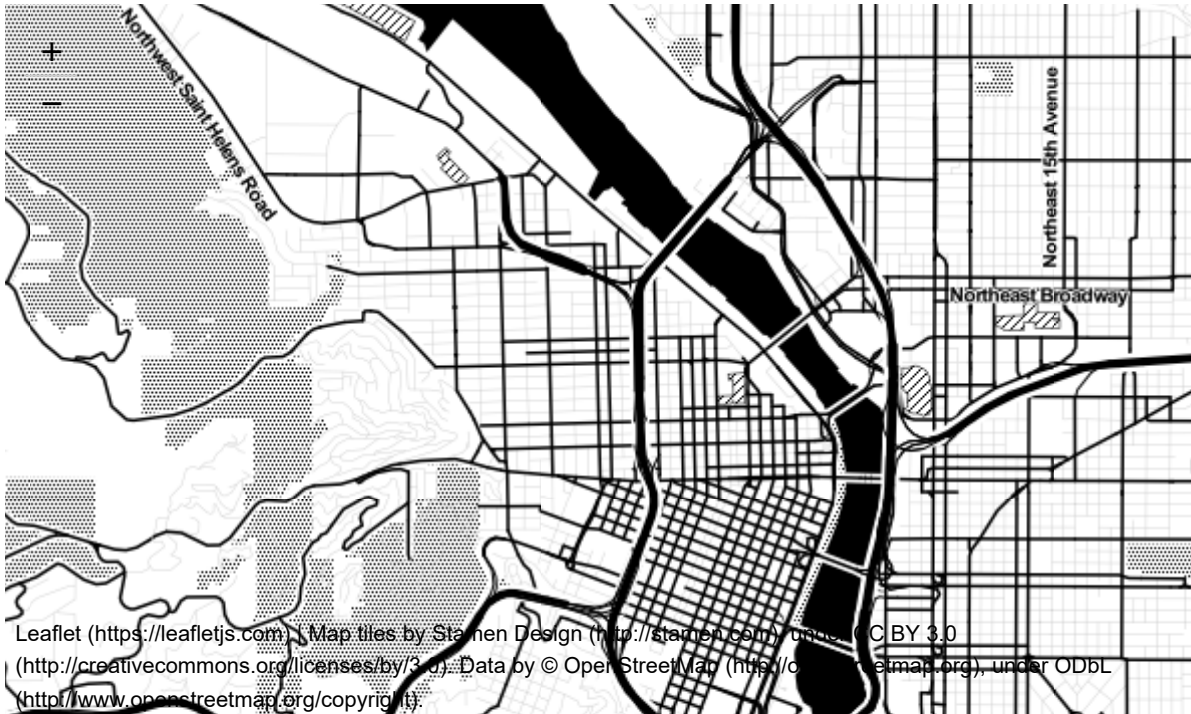
tiles을 값 변경



In [34]:

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

Out[34]:



In [35]:

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

Out [35]:



## Ref

- <https://python-visualization.github.io/folium/modules.html> (<https://python-visualization.github.io/folium/modules.html>)
- <https://dailyheumsi.tistory.com/85> (<https://dailyheumsi.tistory.com/85>)

In [ ]: