

✧ Earthquake Japan since 2023

```
from google.colab import drive
drive.mount('/content/drive')

Drive already mounted at /content/drive; to attempt to forcibly remount, call drive.mount("/content/drive", force_remount=True).
```

Large earthquakes tend to occur within earthquake swarm areas. In other words, earthquake swarms may be a precursor to a future major earthquake. Therefore,the Nankai Trough earthquake, which is predicted to occur in the near future, is likely to be similar.

```
import os
import numpy as np
import pandas as pd
import geopandas as gpd
from shapely.geometry import Point

import folium
from folium import plugins
import seaborn as sns
import matplotlib.pyplot as plt
import IPython
import time
import plotly.express as px

df= gpd.read_file("/content/drive/MyDrive/일본/일본지진/japan_2023-.json")
print(df.columns.tolist())
df[0:3]
```

	id	mag	place	time	updated	tz	url	
0	us7000ludt	4.8	59 km SE of Hasaki, Japan	1706558584349	1706559984040	None	https://earthquake.usgs.gov/earthquakes/eventp...	https://earthquake.usgs
1	us7000lus7	4.6	274 km NNE of Ama, Japan	1706547432940	1706551956040	None	https://earthquake.usgs.gov/earthquakes/eventp...	https://earthquake.usgs
2	us7000luqg	4.6	38 km S of Shizunai- furukawachō, Japan	1706541558867	1706547317040	None	https://earthquake.usgs.gov/earthquakes/eventp...	https://earthquake.usgs

3 rows × 28 columns

```
df=df[['mag', 'geometry']].dropna().reset_index(drop=True)
display(df[0:3])
print(df.iloc[0,1])
```

	mag	geometry
0	4.8	POINT Z (141.32950 35.38030 23.26600)
1	4.6	POINT Z (133.95300 38.47150 436.14600)
2	4.6	POINT Z (142.33240 41.99260 63.57300)

POINT Z (141.3295 35.3803 23.266)

```
df['geometry'] = df['geometry'].apply(lambda geo_str: Point(geo_str))
df['lat'] = df['geometry'].apply(lambda point: point.y)
df['lon'] = df['geometry'].apply(lambda point: point.x)
df['depth'] = df['geometry'].apply(lambda point: point.z)
display(df[0:3])
```

	mag	geometry	lat	lon	depth
0	4.8	POINT Z (141.32950 35.38030 23.26600)	35.3803	141.3295	23.266
1	4.6	POINT Z (133.95300 38.47150 436.14600)	38.4715	133.9530	436.146
2	4.6	POINT Z (142.33240 41.99260 63.57300)	41.9926	142.3324	63.573

The 'scope' property is an enumeration that may be specified as:

- One of the following enumeration values:

['africa', 'asia', 'europe', 'north america', 'south america', 'usa', 'world']

```
from plotly.express import scatter_geo
```

```
scatter_geo(data_frame=df, lat="lat", lon="lon", scope="asia", color="mag", hover_name="mag")
```

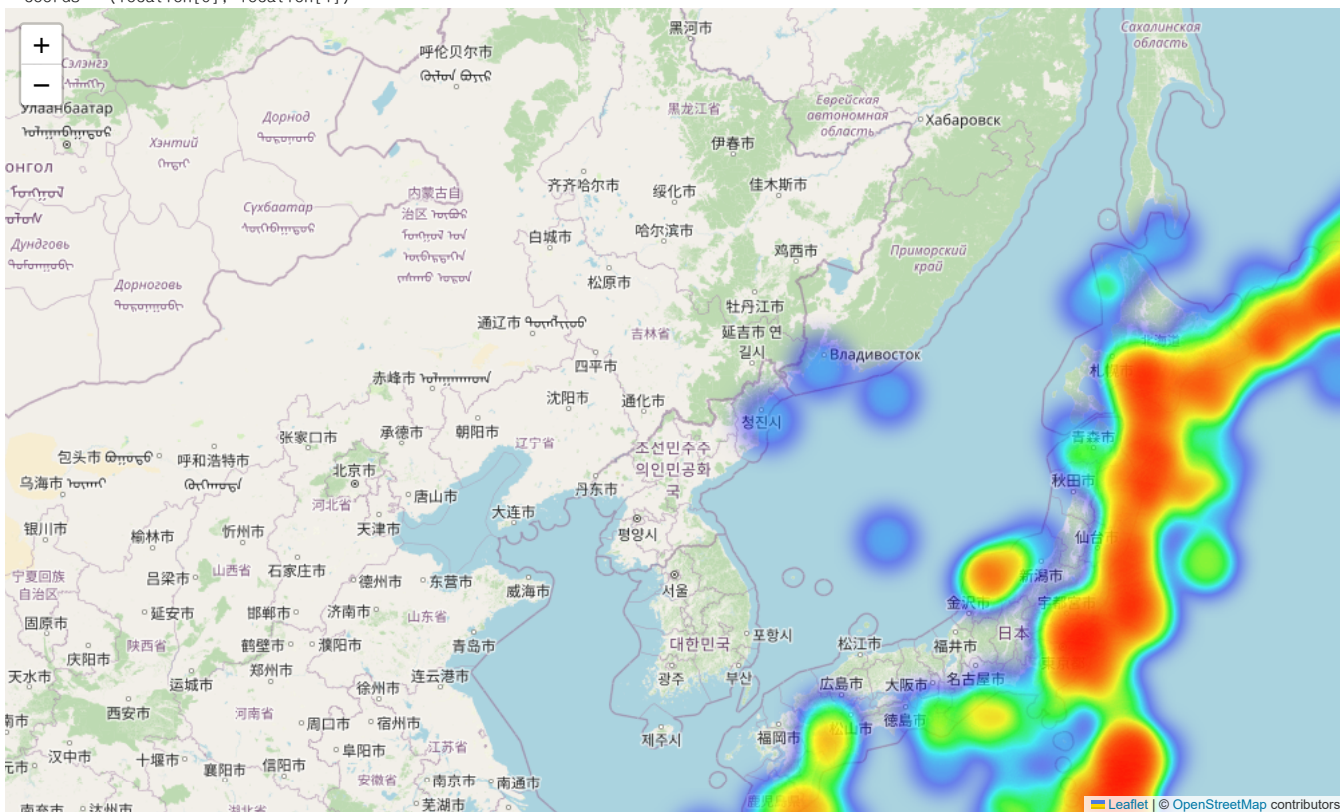
```
l1i=df.iloc[44,2:4]
```

```
eq_map = folium.Map(location=l1i,tiles='OpenStreetMap',zoom_start=5.0,min_zoom=2.0)
```

```
eq_map.add_child(plugins.HeatMap(df.iloc[:,2:4]))
```

```
eq_map
```

 /usr/local/lib/python3.10/dist-packages/folium/utilities.py:86: FutureWarning: Series.__getitem__ treating keys as positions is deprecated. In a



```
from branca.colormap import linear
```

```
eq_map = folium.Map(location=[37.5, 137.3], tiles='OpenStreetMap', zoom_start=5.0, min_zoom=2.0)
```

```
colormap = linear.YlOrRd_05.scale(df['mag'].min(), df['mag'].max())
```

```
for i in range(len(df)):
```

```
    l1i = df.iloc[i,2:4].tolist()
```

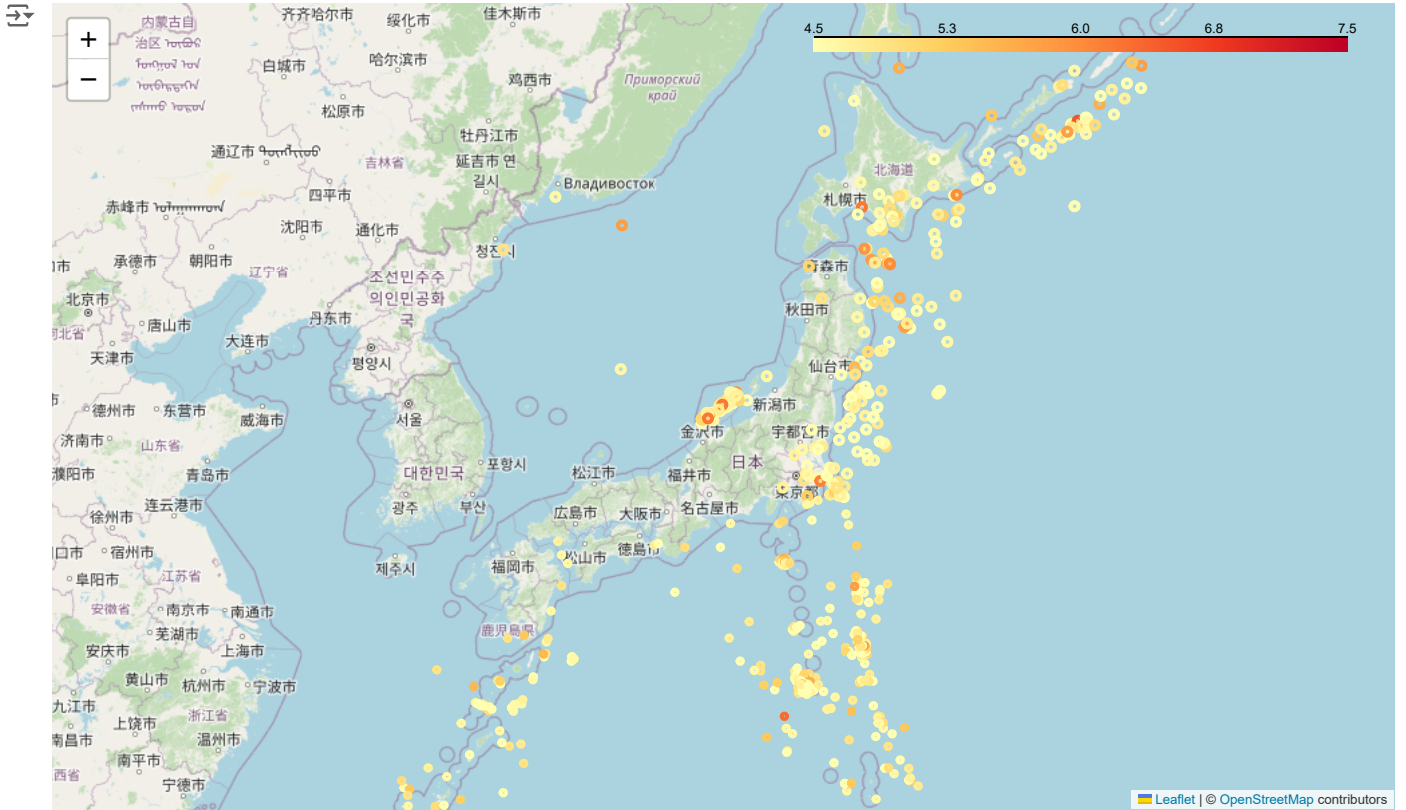
```
    magi = df.iloc[i,0]
```

```
    color = colormap(magi)
```

```
    folium.Circle(location=l1i, radius=10000, color=color, fill_color=color).add_to(eq_map)
```

```
colormap.add_to(eq_map)
```

```
eq_map
```



코딩을 시작하거나 AI로 코드를 생성하세요.

코딩을 시작하거나 AI로 코드를 생성하세요.