I suggest you guys to run kepler on "colab" instead of "JupyterLab"

Install

```
pip install geopandas
```

```
Requirement already satisfied: geopandas in /usr/local/lib/python3.10/dist-packages (0.13.2)
Requirement already satisfied: fiona>=1.8.19 in /usr/local/lib/python3.10/dist-packages (from geopandas) (1.9.4.post1)
Requirement already satisfied: packaging in /usr/local/lib/python3.10/dist-packages (from geopandas) (23.1)
Requirement already satisfied: pandas>=1.1.0 in /usr/local/lib/python3.10/dist-packages (from geopandas) (1.5.3)
Requirement already satisfied: pyproj>=3.0.1 in /usr/local/lib/python3.10/dist-packages (from geopandas) (3.6.0)
Requirement already satisfied: shapely>=1.7.1 in /usr/local/lib/python3.10/dist-packages (from geopandas) (2.0.1)
Requirement already satisfied: attrs>=19.2.0 in /usr/local/lib/python3.10/dist-packages (from fiona>=1.8.19->geopandas) (23.1.0)
Requirement already satisfied: certifi in /usr/local/lib/python3.10/dist-packages (from fiona>=1.8.19->geopandas) (2023.7.22)
Requirement already satisfied: click~=8.0 in /usr/local/lib/python3.10/dist-packages (from fiona>=1.8.19->geopandas) (8.1.6)
Requirement already satisfied: click-plugins>=1.0 in /usr/local/lib/python3.10/dist-packages (from fiona>=1.8.19->geopandas) (1.1.1)
Requirement already satisfied: cligj>=0.5 in /usr/local/lib/python3.10/dist-packages (from fiona>=1.8.19->geopandas) (0.7.2)
Requirement already satisfied: six in /usr/local/lib/python3.10/dist-packages (from fiona>=1.8.19->geopandas) (1.16.0)
Requirement already satisfied: python-dateutil>=2.8.1 in /usr/local/lib/python3.10/dist-packages (from pandas>=1.1.0->geopandas) (2.8.2)
Requirement already satisfied: pytz>=2020.1 in /usr/local/lib/python3.10/dist-packages (from pandas>=1.1.0->geopandas) (2023.3)
Requirement already satisfied: numpy>=1.21.0 in /usr/local/lib/python3.10/dist-packages (from pandas>=1.1.0->geopandas) (1.23.5)
```

```
pip install keplergl
```

```
Requirement already satisfied: click-plugins>=1.0 in /usr/local/lib/python3.10/dist-packages (from fiona>=1.8.19->geopandas>=0.5.0->ke
Requirement already satisfied: cligj>=0.5 in /usr/local/lib/python3.10/dist-packages (from fiona>=1.8.19->geopandas>=0.5.0->keplergl)
Requirement already satisfied: six in /usr/local/lib/python3.10/dist-packages (from fiona>=1.8.19->geopandas>=0.5.0->keplergl) (1.16.0
Requirement already satisfied: jupyter-client in /usr/local/lib/python3.10/dist-packages (from ipykernel>=4.5.1->ipywidgets<8,>=7.0.0
Requirement already satisfied: tornado>=4.2 in /usr/local/lib/python3.10/dist-packages (from ipykernel>=4.5.1->ipywidgets<8,>=7.0.0->i
Requirement already satisfied: setuptools>=18.5 in /usr/local/lib/python3.10/dist-packages (from ipython>=4.0.0->ipywidgets<8,>=7.0.0
Collecting jedi>=0.16 (from ipython>=4.0.0->ipywidgets<8,>=7.0.0->keplergl)
  Using cached jedi-0.19.0-py2.py3-none-any.whl (1.6 MB)
Requirement already satisfied: decorator in /usr/local/lib/python3.10/dist-packages (from ipython>=4.0.0->ipywidgets<8,>=7.0.0->keple
Requirement already satisfied: pickleshare in /usr/local/lib/python3.10/dist-packages (from ipython>=4.0.0->ipywidgets<8,>=7.0.0->kepl
Requirement already satisfied: prompt-toolkit!=3.0.0,!=3.0.1,<3.1.0,>=2.0.0 in /usr/local/lib/python3.10/dist-packages (from ipython>=
Requirement already satisfied: pygments in /usr/local/lib/python3.10/dist-packages (from ipython>=4.0.0->ipywidgets<8,>=7.0.0->keplerg
Requirement already satisfied: backcall in /usr/local/lib/python3.10/dist-packages (from ipython>=4.0.0->ipywidgets<8,>=7.0.0->keplerg
Requirement already satisfied: matplotlib-inline in /usr/local/lib/python3.10/dist-packages (from ipython>=4.0.0->ipywidgets<8,>=7.0.0
Requirement already satisfied: pexpect>4.3 in /usr/local/lib/python3.10/dist-packages (from ipython>=4.0.0->ipywidgets<8,>=7.0.0->kepl
Requirement already satisfied: notebook>=4.4.1 in /usr/local/lib/python3.10/dist-packages (from widgetsnbextension~=3.6.0->ipywidgets<
Requirement already satisfied: parso<0.9.0,>=0.8.3 in /usr/local/lib/python3.10/dist-packages (from jedi>=0.16->ipython>=4.0.0->ipywid
Requirement already satisfied: jinja2 in /usr/local/lib/python3.10/dist-packages (from notebook>=4.4.1->widgetsnbextension~=3.6.0->ipy
Requirement already satisfied: pyzmq>=17 in /usr/local/lib/python3.10/dist-packages (from notebook>=4.4.1->widgetsnbextension~=3.6.0-
Requirement already satisfied: argon2-cffi in /usr/local/lib/python3.10/dist-packages (from notebook>=4.4.1->widgetsnbextension~=3.6.0
Requirement already satisfied: jupyter-core>=4.6.1 in /usr/local/lib/python3.10/dist-packages (from notebook>=4.4.1->widgetsnbextensio
Requirement already satisfied: nbformat in /usr/local/lib/python3.10/dist-packages (from notebook>=4.4.1->widgetsnbextension~=3.6.0->i
Requirement already satisfied: nbconvert in /usr/local/lib/python3.10/dist-packages (from notebook>=4.4.1->widgetsnbextension~=3.6.0-
Requirement already satisfied: nest-asyncio>=1.5 in /usr/local/lib/python3.10/dist-packages (from notebook>=4.4.1->widgetsnbextension
Requirement already satisfied: Send2Trash>=1.8.0 in /usr/local/lib/python3.10/dist-packages (from notebook>=4.4.1->widgetsnbextension
Requirement already satisfied: terminado>=0.8.3 in /usr/local/lib/python3.10/dist-packages (from notebook>=4.4.1->widgetsnbextension~
Requirement already satisfied: prometheus-client in /usr/local/lib/python3.10/dist-packages (from notebook>=4.4.1->widgetsnbextension
```

```
    Stored in directory: /root/.cache/pip/wheels/e5/e9/db/b8b8f2fcb5cb1cd7268d88959fce267bc0069/cb8f50dcd0d9
    Successfully built keplergl
    Installing collected packages: traittypes, jedi, keplergl
```

Import

```python
import pandas as pd
import geopandas as gpd
from keplergl import KeplerGl
```
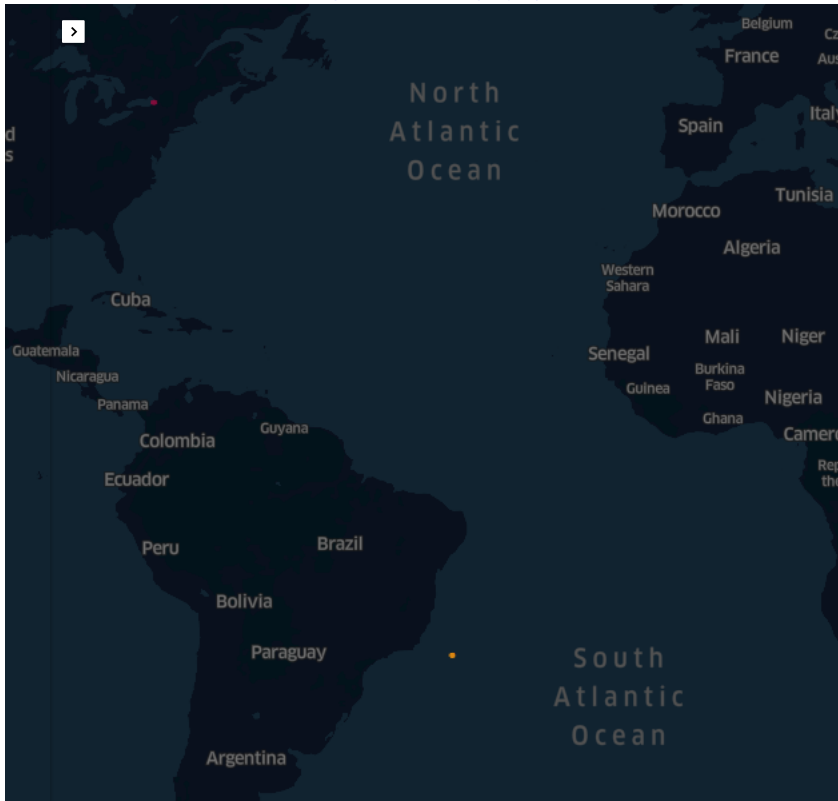
Third Party Jupyter widget for Colab

```python
from google.colab import output
output.enable_custom_widget_manager()
```

Base map

```python
m = KeplerGl(height=600, width=800)
m
```

User Guide: https://docs.kepler.gl/docs/keplergl-jupyter
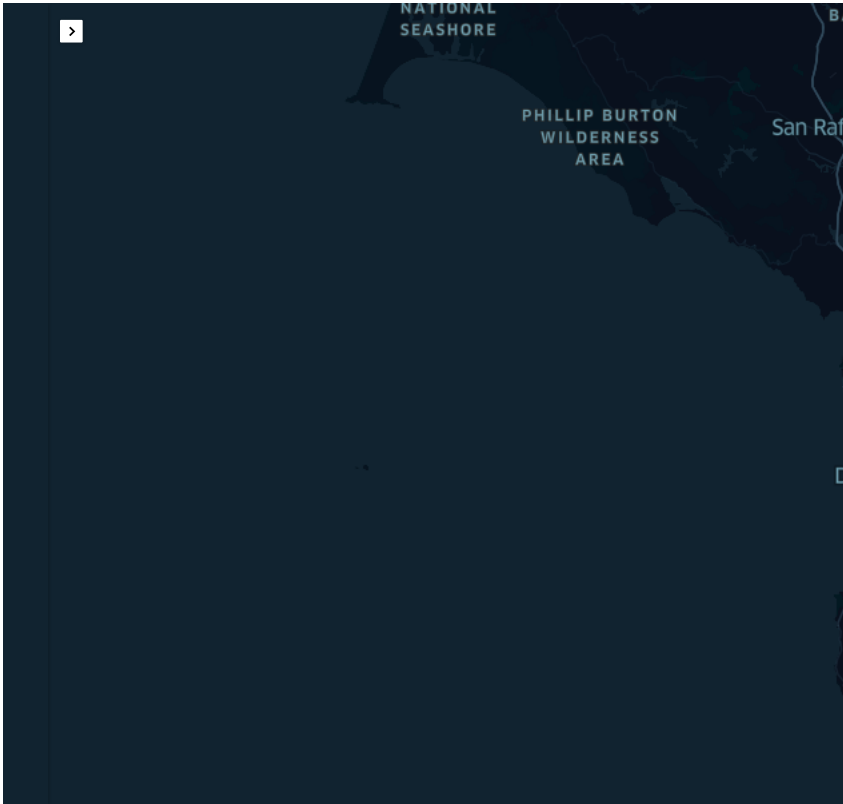

Data frame

```python
df = pd.DataFrame(
    {
        "City": ["Shanghai", "Sao paulo", "Cairo", "London", "Toronto", "Sydney"],
        "Country": ["China", "Brazil", "Egypt", "England", "Canada", "Australia"],
        "Latitude": [31.045556, -23.473293, 30.05, 51.514125, 43.666667, -33.861481],
        "Longitude": [121.399722, -46.665803, 31.25, -0.093689, -79.416667, 151.205475],
        "Population": [14608512, 10021437, 7734602, 7421228, 4612187, 4394585],
    }
)
```

```python
df
```

	City	Country	Latitude	Longitude	Population
0	Shanghai	China	31.045556	121.399722	14608512
1	Sao paulo	Brazil	-23.473293	-46.665803	10021437
2	Cairo	Egypt	30.050000	31.250000	7734602
3	London	England	51.514125	-0.093689	7421228

locate cities on the map

```
m.add_data(data=df, name='cities')
m
```


P.S: for customizing use the Kepler tabs

GeoDataFrame

```
import pandas as pd
import geopandas as gpd
from keplergl import KeplerGl


parking_df = pd.read_csv("parking_violations_2015.csv")
parking_df.head()
```

	anon_ticket_number	issue_datetime	anon_plate_id	violation_desc	fine	issuin
0	4674379	01/01/2015 00:00	934383	BLOCKNG MASS TRANSIT	101	
1	4707189	01/01/2015 00:00	1065037	BLOCKNG MASS TRANSIT	101	
2	4584526	01/01/2015 00:01	1262953	SIDEWALK CC	76	

```
parking_df.shape
```

```
(119910, 8)
```

```
parking_df.dropna(subset=["lat", "lon"], how='all', inplace=True)
```

```
parking_df.shape
```

    (107710, 8)

```
gdf = gpd.GeoDataFrame(parking_df, geometry=gpd.points_from_xy(parking_df.lon, parking_df.lat), crs="EPSG:4326")
gdf.head()
```
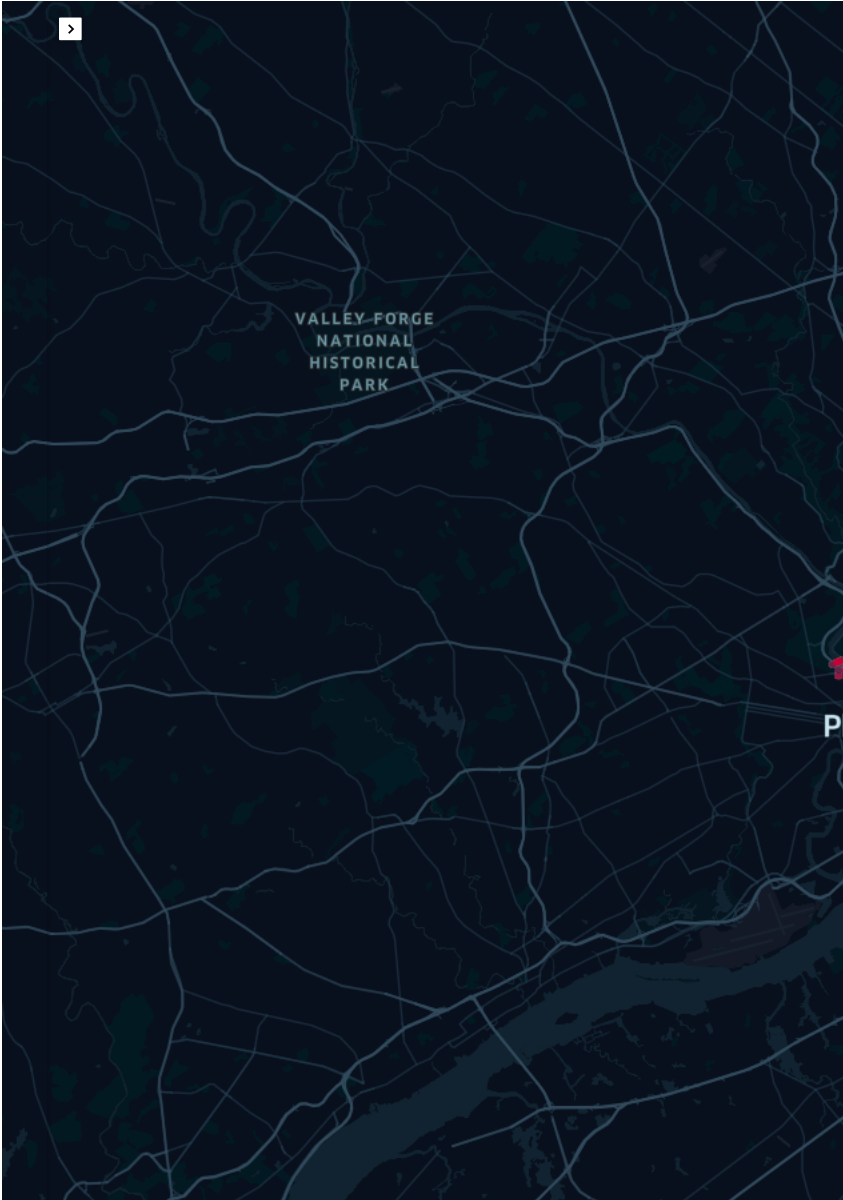
	anon_ticket_number	issue_datetime	anon_plate_id	violation_desc	fine	issuin
0	4674379	01/01/2015 00:00	934383	BLOCKNG MASS TRANSIT	101	
1	4707189	01/01/2015 00:00	1065037	BLOCKNG MASS TRANSIT	101	

```
gdf.to_file("parking_2015.geojson",  driver='GeoJSON')
```

```
m = KeplerGl(height=900)
m.add_data(data=gdf, name='parking_points')
m
```

User Guide: https://docs.kepler.gl/docs/keplergl-jupyter

```
import geopandas as gpd
from keplergl import KeplerGl
```

3D visualize

```
gdf = gpd.read_file("parking_2015.geojson")
gdf.head()
```

	anon_ticket_number	issue_datetime	anon_plate_id	violation_desc	fine	issuin
0	4674379	01/01/2015 00:00	934383	BLOCKNG MASS TRANSIT	101	
1	4707189	01/01/2015 00:00	1065037	BLOCKNG MASS TRANSIT	101	

```
m = KeplerGl(height=900)
m.add_data(data=gdf, name='3dHexagon')
m
```

User Guide: https://docs.kepler.gl/docs/keplergl-jupyter

Animate

```
import geopandas as gpd
import pandas as pd
from keplergl import KeplerGl
```

```
gdf = gpd.read_file("parking_2015.geojson", parse_dates=True)
gdf.head()
```

	anon_ticket_number	issue_datetime	anon_plate_id	violation_desc	fine	issuin
0	4674379	01/01/2015 00:00	934383	BLOCKNG MASS TRANSIT	101	
1	4707189	01/01/2015 00:00	1065037	BLOCKNG MASS TRANSIT	101	

```
gdf["issue_datetime"] = pd.to_datetime(gdf["issue_datetime"], dayfirst=True)
```
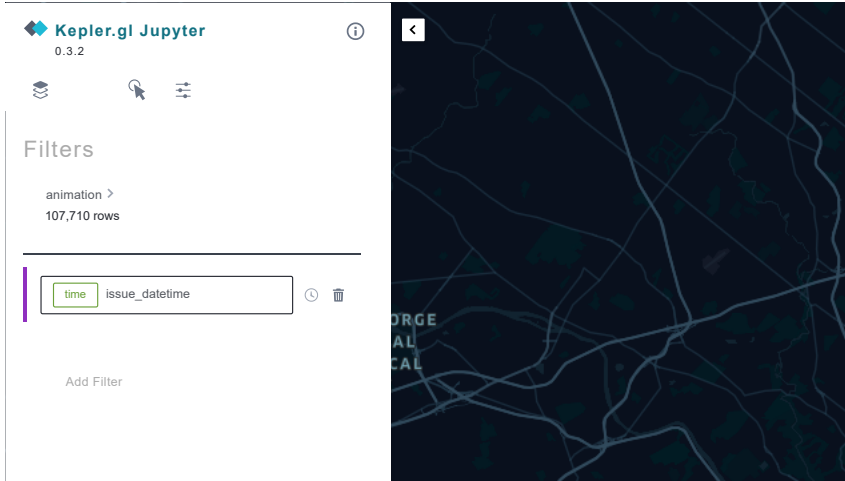
```
gdf.sample(10)
```

	anon_ticket_number	issue_datetime	anon_plate_id	violation_desc	fine	i
104743	4633708	2015-01-31 10:30:00	1328468	METER EXPIRED CC	36	
4252	4575171	2015-01-03 10:22:00	1540074	METER EXPIRED CC	36	
58033	4622972	2015-01-18 09:00:00	985476	BLOCKING DRIVEWAY	51	
43555	4683288	2015-01-14 16:52:00	1277624	METER EXPIRED	26	
91629	4514288	2015-01-28 12:45:00	1398166	METER EXPIRED CC	36	

```
m = KeplerGl(height=900)
m.add_data(data=gdf, name='animation')
m
```

User Guide: https://docs.kepler.gl/docs/keplergl-jupyter

◆◆ **Kepler.gl Jupyter**   ⓘ
0.3.2

Filters

animation ›
107,710 rows

┌─────────────────────────────────────┐
│ time issue_datetime   ⏱ 🗑   │
└─────────────────────────────────────┘

Add Filter

Choropleth Map

```
import geopandas as gpd
from keplergl import KeplerGl
neighborhoods = gpd.read_file("neighbrhoods.geojson")
neighborhoods.head()
```

	NAME	LISTNAME	MAPNAME	Shape_Leng	Shape_Area	NUMPOINTS
0	BRIDESBURG	Bridesburg	Bridesburg	27814.546521	4.458626e+07	149.0
1	BUSTLETON	Bustleton	Bustleton	48868.458365	1.140504e+08	228.0

```
neighborhoods = neighborhoods.to_crs("EPSG:4326")
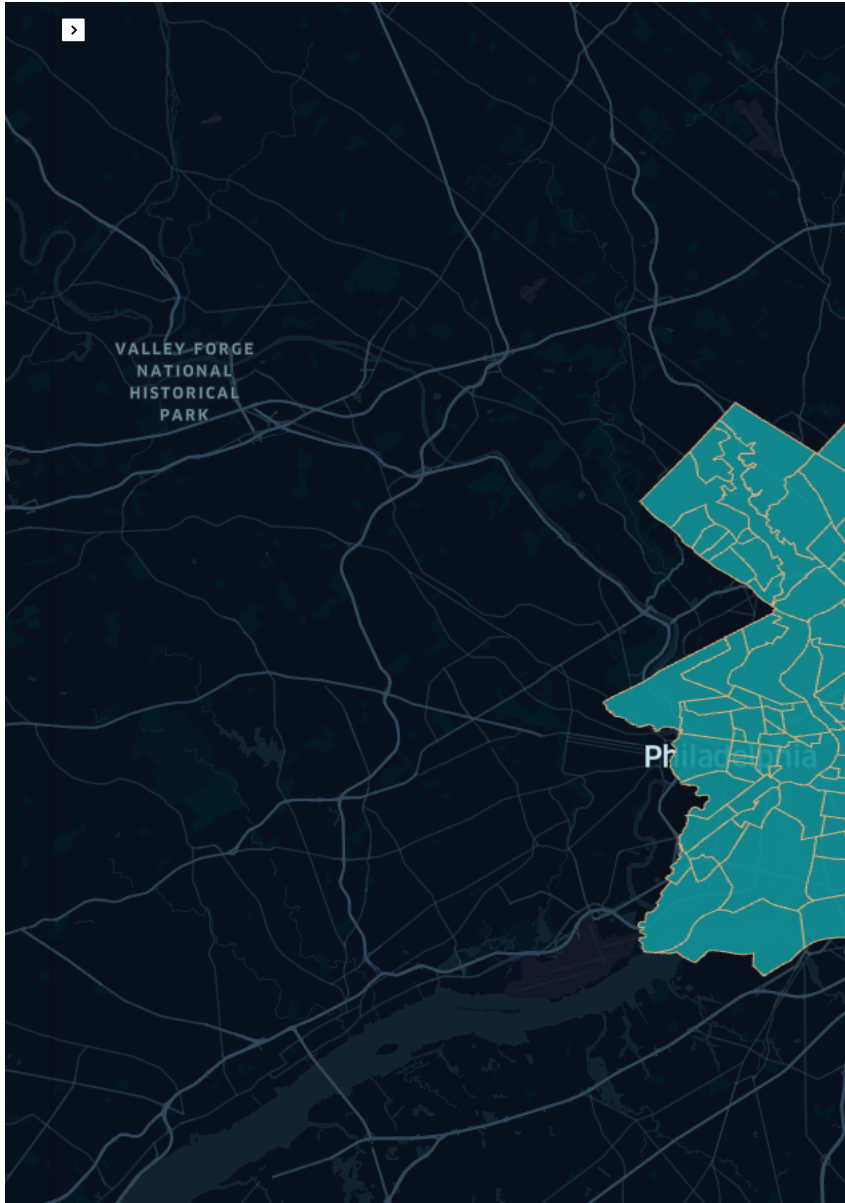```

```
neighborhoods.crs
```

```
<Geographic 2D CRS: EPSG:4326>
Name: WGS 84
Axis Info [ellipsoidal]:
- Lat[north]: Geodetic latitude (degree)
- Lon[east]: Geodetic longitude (degree)
Area of Use:
- name: World.
- bounds: (-180.0, -90.0, 180.0, 90.0)
Datum: World Geodetic System 1984 ensemble
- Ellipsoid: WGS 84
- Prime Meridian: Greenwich
```

```
m = KeplerGl(height=900)
m.add_data(data=neighborhoods, name='choropleth_map')
m
```

User Guide: https://docs.kepler.gl/docs/keplergl-jupyter

0s   completed at 2:03 AM