

LeafletMap

April 28, 2020

1 GIS Data Processing with Non-ESRI Tools

1.1 Toolkits

- Jupyter Notebooks
- Leaflet
- Python
- ipyleaflet, ipywidgets
- geopandas

1.2 Steps

Export data from Survey123 to Shapefile:

Start Jupyter:

jupyter notebook LeafletMap.ipynb

Import Files

```
In [1]: from ipyleaflet import Map, Marker, basemaps, GeoData, GeoJSON, Popup
        from ipywidgets import HTML
        import fiona
        import geopandas
        import json
```

Anonymize Data

```
geo_frame = geopandas.read_file('survey.shp')
anon_frame = geo_frame[['space_name', 'space_publ', 'space_acti', 'space_freq', 'space_last', 'spa
anon_frame.to_file("anon_results.json", driver="GeoJSON")
```

```
In [2]: activity_map = {'space_activity_run': 'Run',
                        'space_activity_bicycle': 'Bicycle',
                        'space_activity_walk': 'Walk',
                        'space_activity_hike': 'Hike',
                        'space_activity_just_be': 'Just Be',
                        'other': 'Other'}
```

Load Dataset and Create Leaflet Map

```

In [3]: with open('anon_results.json', 'r') as f:
        data = json.load(f)

        #Define map center as Scott's Addition
        center = (37.566272, -77.47060)

        #Create map
        m = Map(center = center, zoom = 12)#, basemap= basemaps.Esri.WorldTopoMap)

        for row in data['features']:
            loc = (row['geometry']['coordinates'][1], row['geometry']['coordinates'][0])
            message1 = HTML()
            message1.value = '{} - {}'.format(row['properties']['space_name'],
                                                activity_map[row['properties']['space_acti']].spl

            #marker = Marker(location=loc)
            popup = Popup(location=loc,
                           child=message1,
                           close_button=False,
                           auto_close=False,
                           close_on_escape_key=False
            )
            m.add_layer(popup)

        #geo_json = GeoJSON(data=data, style={'tooltip': 'foo'})

        display(m)

Map(center=[37.566272, -77.4706], controls=(ZoomControl(options=['position', 'zoom_in_text', '2

```