**# Overview**

The objective of the task are is to create a interactive map using Leaflet.js API, a map was populated with GeoJSON earthquake data from a URL. Each earthquake with a magnitude greater than 4.5 is visually represented with a circle and color, and a larger magnitude will have a larger diameter circle. In general, the generated map comprise of three different map overlays, all recorded earthquake data over the 7 days. A Users can choose to view the map in street view, satellite street view, or light view.

**# Dependencies**

- JavaScript

    - D3.js

    - Leaflet.js

- Mapbox

- GeoJSON data

    - USGS All Earthquakes in the Last 7 Days

    - USGS Earthquake with > 4.5 Magnitude in the Last 7 Days

    - Tectonic Plates Data

**# Results**

Salient features of the Maps:

1. Each earthquake was represented by a circle and color that matches certain features of the earthquake; the higher the magnitude of the earthquake the larger diameter and darker the color

2. A control feature that allows the user to change visible layers

3. Each earthquake was enabled with a pop up feature that displays pertinent information relevant to the earthquake

Code was written in Javascript (see challenge\_final.js) to create the required map, and script was written in HTML so the map can load and be viewed on a webpage (see index.html).

1. Use mapbox tile layer to create the background of the map and create the map object with center, zoom level and default layer.

2. Add control to the map that will allow the user to change visible layers.

3. Use d3.json() callback method to make a call to the tectonic plate and earthquake data

4. Use Leaflet.js Application Programming Interface (API) to populate the geographical map with GeoJSON earthquake and tectonic plate data from a URL.

5. Write functions to style the required map layers.

6. Write code to create a legend for the earthquake data.

Deliverable 1:

Using JavaScript, Leaflet.js, and geoJSON data, add tectonic plate data using d3.json(), add the data using the geoJSON() layer, set the tectonic plate LineString data to stand out on the map, and add the tectonic plate data to the overlay object with the earthquake data.

Deliverable 2:

Using JavaScript, Leaflet.js, and geoJSON data, add major earthquake data to the map using d3.json(), and a color and set the radius of the circle based on the magnitude of earthquake, and add a popup marker for each earthquake that displays the magnitude and location of the earthquake using the GeoJSON layer, geoJSON().

Deliverable 3:

Using JavaScript and Leaflet.js a third map style was added to the earthquake map.