

Leaflet Homework - Visualizing Data with Leaflet

Background



Welcome to the United States Geological Survey, or USGS for short! The USGS is responsible for providing scientific data about natural hazards, the health of our ecosystems and environment; and the impacts of climate and land-use change. Their scientists develop new methods and tools to supply timely, relevant, and useful information about the Earth and its processes. As a new hire, you will be helping them out with an exciting new project!

The USGS is interested in building a new set of tools that will allow them visualize their earthquake data. They collect a massive amount of data from all over the world each day, but they lack a meaningful way of displaying it. Their hope is that being able to visualize their data will allow them to better educate the public and other government organizations (and hopefully secure more funding..) on issues facing our planet.

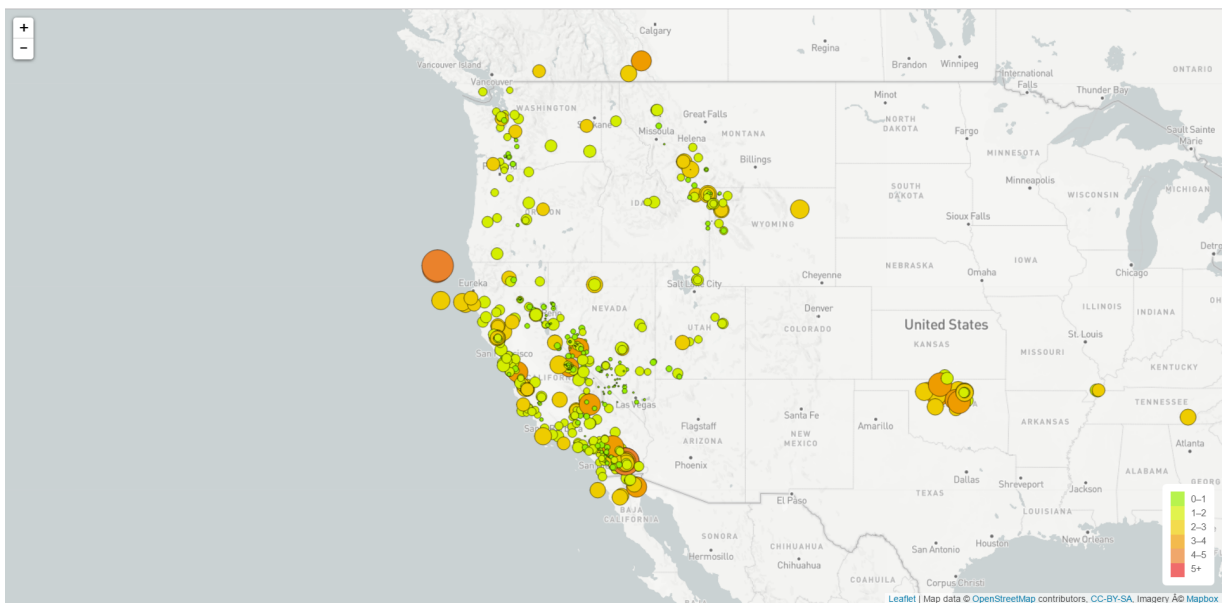
Before You Begin

1. Create a new repository for this project called `leaflet-challenge` . **Do not add this homework to an existing repository.**
2. Clone the new repository to your computer.

3. Inside your local git repository, create a directory for the Leaflet challenge. Use the folder names to correspond to the challenges: **Leaflet-Step-1** and **Leaflet-Step-2**.
4. This homeworks utilizes both **html** and **Javascript** so be sure to add all the necessary files. These will be the main files to run for analysis.
5. Push the above changes to GitHub or GitLab.


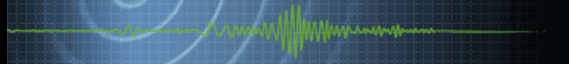
Your Task

Level 1: Basic Visualization



Your first task is to visualize an earthquake data set.

1. Get your data set

Earthquake Hazards Program

Feeds & Notifications

Real-time Feeds

ATOM
KML
Spreadsheet
QuakeML

Real-time Notifications

Earthquake Notification Service
Tweet Earthquake Dispatch

For Developers

API Documentation - EQ Catalog
GeoJSON Summary
GeoJSON Detail

GeoJSON Summary Format

Description

GeoJSON is a format for encoding a variety of geographic data structures. A GeoJSON object may represent a geometry, a feature, or a collection of features. GeoJSON uses the [JSON standard](#). The GeoJSONP feed uses the same JSON response, but the GeoJSONP response is wrapped inside the function call, eqfeed_callback. See the [GeoJSON site](#) for more information.

This feed adheres to the USGS Earthquakes [Feed Life Cycle Policy](#).

Usage

GeoJSON is intended to be used as a programatic interface for applications.

Output

```
{
  "type": "FeatureCollection",
  "metadata": {
    "generated": Long Integer,
    "url": String,
    "title": String,
    "api": String,

```

Feeds

Past Hour

Updated every 5 minutes.

- [Significant Earthquakes](#)
- [M4.5+ Earthquakes](#)
- [M2.5+ Earthquakes](#)
- [M1.0+ Earthquakes](#)
- [All Earthquakes](#)

Past Day

Updated every 5 minutes.

- [Significant Earthquakes](#)
- [M4.5+ Earthquakes](#)
- [M2.5+ Earthquakes](#)
- [M1.0+ Earthquakes](#)
- [All Earthquakes](#)

Past 7 Days

The USGS provides earthquake data in a number of different formats, updated every 5 minutes. Visit the [USGS GeoJSON Feed](#) page and pick a data set to visualize. When you click on a data set, for example 'All Earthquakes from the Past 7 Days', you will be given a JSON representation of that data. You will be using the URL of this JSON to pull in the data for our visualization.

```
{
  "type": "FeatureCollection",
  "metadata": {
    "generated": 1475036500000,
    "url": "http://earthquake.usgs.gov/earthquakes/feed/v1.0/summary/all_day.geojson",
    "title": "USGS All Earthquakes, Past Day",
    "status": 200,
    "api": "1.5.2",
    "count": 358
  },
  "features": [
    {
      "type": "Feature",
      "properties": {
        "mag": 1.29,
        "place": "13km ENE of Borrego Springs, CA",
        "time": 1475035516570,
        "updated": 1475035737999,
        "tz": -420,
        "url": "http://earthquake.usgs.gov/earthquakes/eventpage/ci37703760",
        "detail": "http://earthquake.usgs.gov/earthquakes/feed/v1.0/detail/ci37703760.geojson",
        "felt": null,
        "cdi": null,
        "mmi": null,
        "alert": null,

```

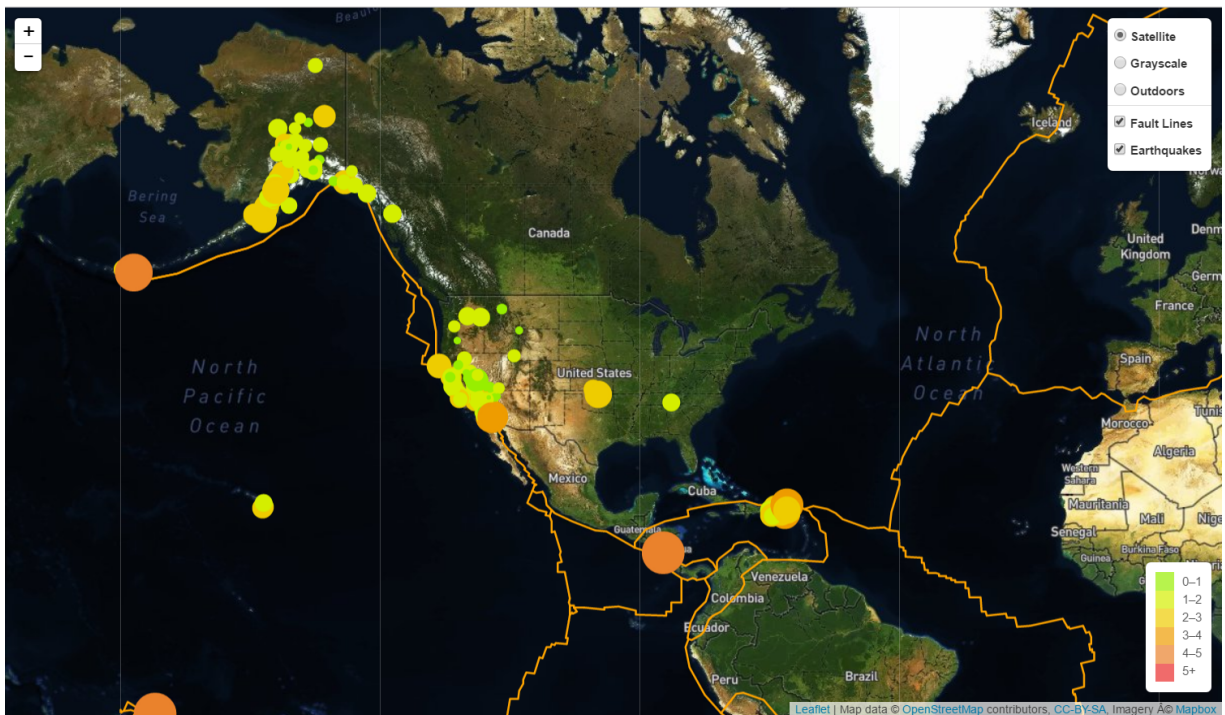
2. Import & Visualize the Data

Create a map using Leaflet that plots all of the earthquakes from your data set based on their longitude and latitude.

- Your data markers should reflect the magnitude of the earthquake in their size and color. Earthquakes with higher magnitudes should appear larger and darker in color.
- Include popups that provide additional information about the earthquake when a marker is clicked.
- Create a legend that will provide context for your map data.

- Your visualization should look something like the map above.

Level 2: More Data (Optional)



The USGS wants you to plot a second data set on your map to illustrate the relationship between tectonic plates and seismic activity. You will need to pull in a second data set and visualize it along side your original set of data. Data on tectonic plates can be found at <https://github.com/fraxen/tectonicplates>.

In this step we are going to..

- Plot a second data set on our map.
- Add a number of base maps to choose from as well as separate out our two different data sets into overlays that can be turned on and off independently.
- Add layer controls to our map.

Assessment

Your final product will be assessed on the following metrics:

- Completion of assigned tasks
- Visual appearance

- Professionalism

Good luck!

Copyright

Trilogy Education Services © 2019. All Rights Reserved.