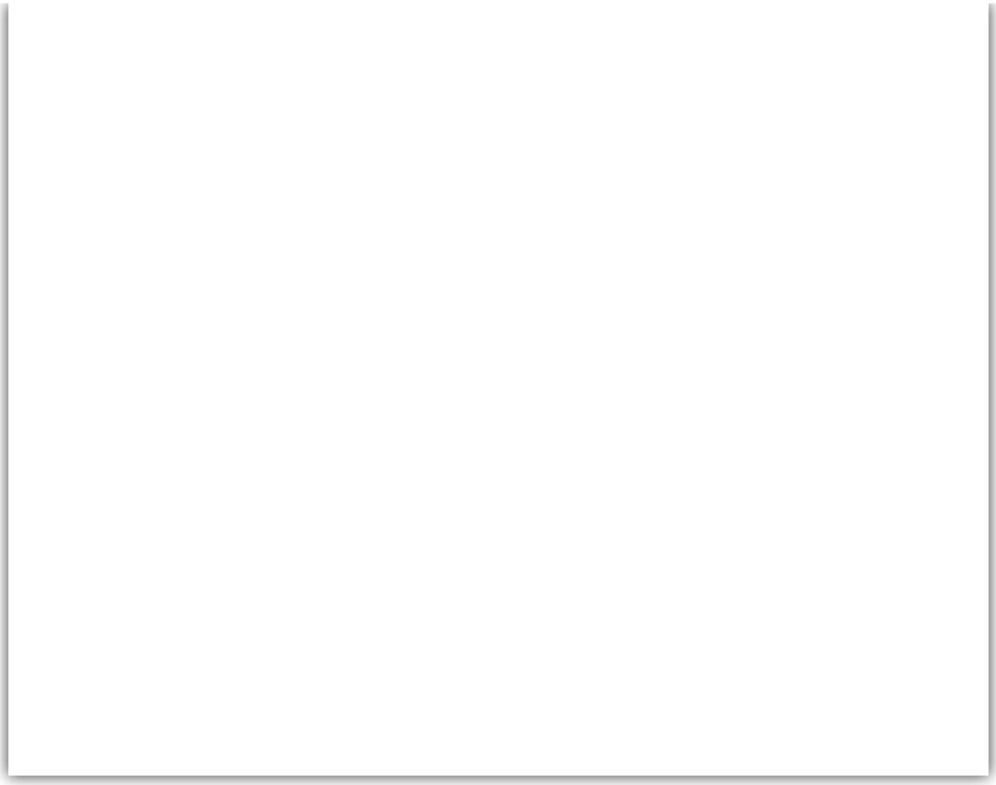
13.6.1 Add Earthquake Data to a Map

Now that you know how to access GeoJSON data, parse the data, and add it to a map, Sadhana would like you to map all recorded earthquakes in the past seven days. Once you get the data, you'll add some features to the map to showcase the severity of earthquakes for viewers.

As before, we need to set up a folder structure for our project in a new branch. Create a branch called "Earthquakes_past7days." Copy the folders and files from one of your previous branches and add them to the Earthquakes_past7days folder.



First, Sadhana wants you to rename the `logic.js` file to `logicStep1.js`. This way, each step has its own `logic.js` file that can be used by other interns in the future.

Now we'll edit our `logicStep1.js` file to create a map with all recorded earthquakes from the past seven days.

First, apply the `streets` and `satelliteStreets` map styles used for the GeoJSON polygon mapping. Change the text for the maps on the base layer to read as "Streets" and "Satellite" to look like the following:

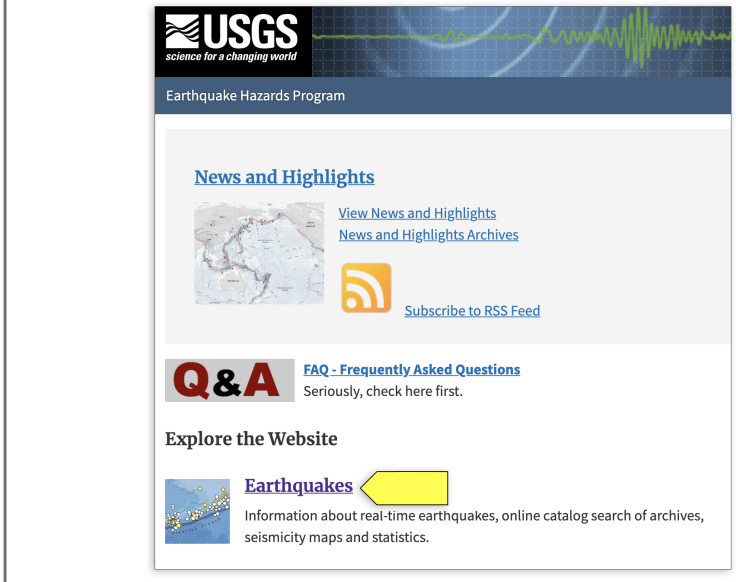
```
// Create a base layer that holds both maps.
let baseMaps = {
  "Streets": streets,
  "Satellite": satelliteStreets
};
```

Change the center of our map to the geographic center of the United States using the coordinates `[39.5, -98.5]`, with a zoom level of 3 and default layer `streets`. Our `logicStep1.js` file should look like the following:

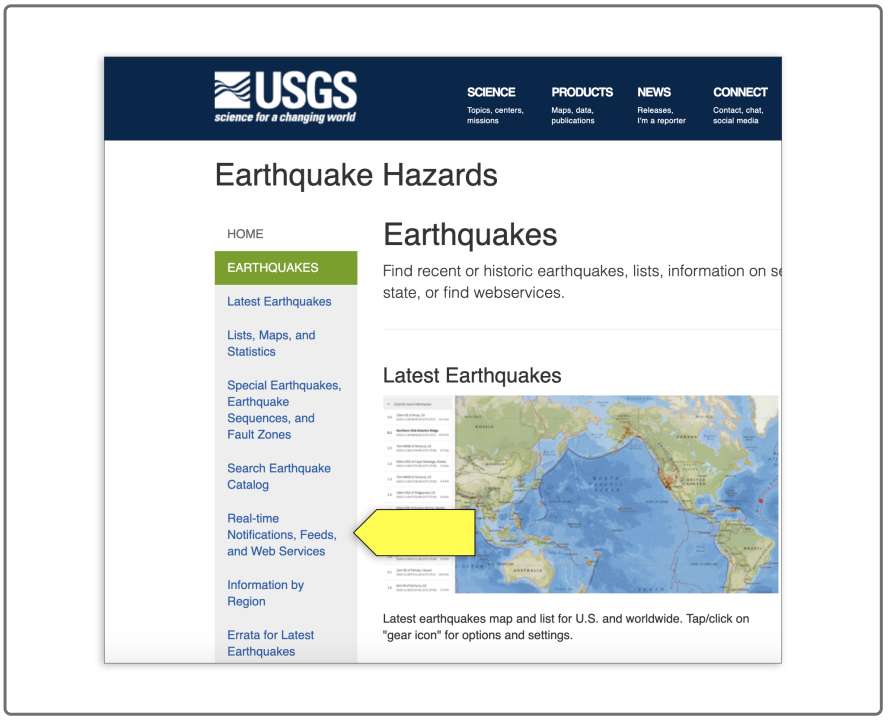
```
// Add console.log to check to see if our code is working.
console.log("working");

// We create the tile layer that will be the background of our map.
let streets = L.tileLayer('https://api.mapbox.com/styles/v1/mapbox/streets-v11/tiles/{z}/{x}/{y}?access_token={accessToken}', {
    attribution: 'Map data &copy; <a href="https://www.openstreetmap.org/">OpenStreetMap</a> contributors, <a href="https://creativecommons.or
    maxZoom: 18,
    accessToken: API_KEY
});

// We create the tile layer that will be the background of our map.
let satelliteStreets = L.tileLayer('https://api.mapbox.com/styles/v1/mapbox/satellite-streets-v11/tiles/{z}/{x}/{y}?access_token={accessToke
    attribution: 'Map data &copy; <a href="https://www.openstreetmap.org/">OpenStreetMap</a> contributors, <a href="https://creativecommons.or
    maxZoom: 18,
    accessToken: API_KEY
});

// Create a base layer that holds both maps.
let baseMaps = {
    "Streets": streets,
    "Satellite": satelliteStreets
};

// Create the map object with center, zoom level and default layer.
let map = L.map('mapid', {
    center: [39.5, -98.5],
    zoom: 3,
    layers: [streets]
});

// Pass our map layers into our layer control and add the layer control to the map.
L.control.layers(baseMaps).addTo(map);
```

Add the USGS URL for earthquake data by following these steps:

1. From the **USGS home page (https://earthquake.usgs.gov/)** click the **Earthquakes (https://www.usgs.gov/natural-hazards/earthquake-hazards/earthquakes)** link:

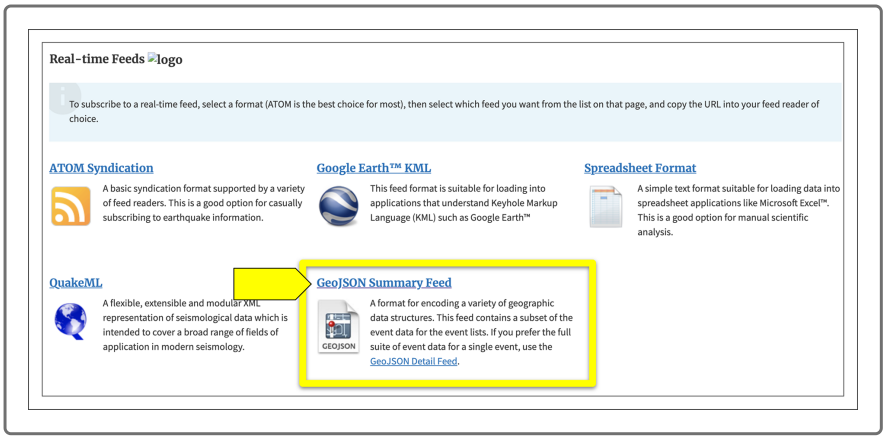


2. Next, click the **Real-time Notifications, Feeds, and Web Services (https://earthquake.usgs.gov/earthquakes/feed/)** link:

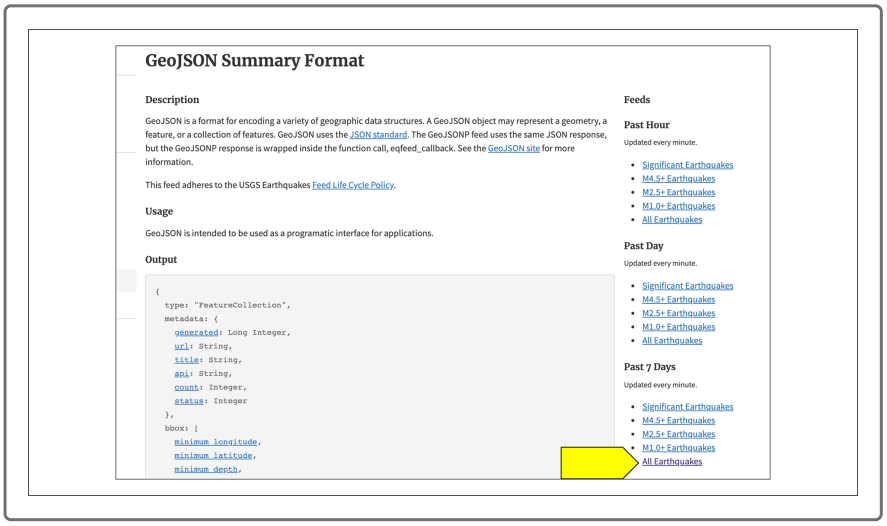


3. Scroll down until you see "GeoJSON Summary Feed".

4. Click the **GeoJSON Summary Feed (https://earthquake.usgs.gov/earthquakes/feed/v1.0/geojson.php)** link:



5. On the right-hand side, click the All Earthquakes link under the "Past 7 Days" heading:



Nice work! The GeoJSON data will launch in your browser:

If we look closer at the geometry object, we'll see an additional data point in the coordinates object, 3.91, which is the depth of the earthquake in kilometers:

```
- geometry: {
    type: "Point",
  - coordinates: [
      -116.3505,
      33.9515,
      3.91
    ]
  };
```

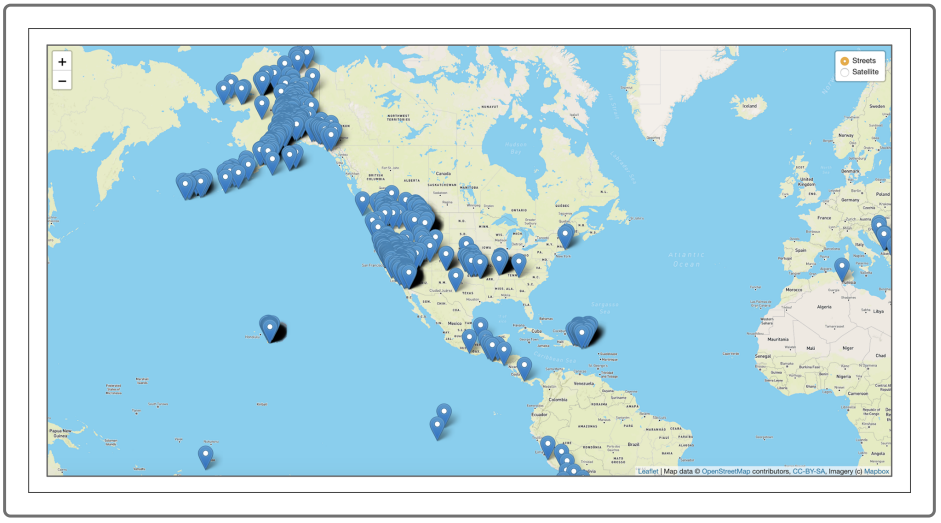
NOTE

> For more information on earthquake depth and other terms, see the **Event Terms (https://earthquake.usgs.gov/data/comcat/data-eventterms.php)**.

Copy the URL for the earthquake JSON data recorded for the past seven days, and add it in place of the previous URL in the `d3.json()` method. It should look like the following:

```
// Retrieve the earthquake GeoJSON data.
d3.json("https://earthquake.usgs.gov/earthquakes/feed/v1.0/summary/all_week.
  // Creating a GeoJSON layer with the retrieved data.
  L.geoJSON(data).addTo(map);
});
```

After saving the `logicStep1.js` file and opening the `index.html` file in your browser, the map should look like the following. Make sure you are referring to the correct `logic` file in your `index.html` file:



Great job adding the earthquake data to our maps!

ADD/COMMIT/PUSH

Add, commit, and push your changes to your Earthquakes_past7days branch.

Let's make this data visually interesting by changing the marker to a circle with a radius representing the earthquake's magnitude, and then we'll style each earthquake data point.