Follow the instructions below and the numbered comments in the starter code to complete Deliverable 1.

1. Create a new folder on your Mapping\_Earthquakes repository and name it "Earthquake\_Challenge."
2. Copy the folders and files from your Earthquakes\_past7days branch and add them to the Earthquake\_Challenge folder. The folder should have this structure:
   * Earthquake\_Challenge folder
     + index.html
     + static
       - css
         * style.css
       - js
         * config.js
         * logic.js
3. Download the tectonic\_plate\_starter\_logic.js file, add it to your js folder, and rename it challenge\_logic.js.
4. In Step 1, add a second layer group variable for the tectonic plate data.
5. In Step 2, add a reference to the tectonic plate data to the overlay object.
6. In Step 3, using d3.json() callback method, make a call to the tectonic plate data using the GeoJSON/PB2002\_boundaries.json data from this [GitHub repository (Links to an external site.)](https://github.com/fraxen/tectonicplates) for the tectonic plate data. You’ll need to log into GitHub to access the GeoJSON data.
7. Inside the d3.json() method do the following:
   * Pass the tectonic plate data to the geoJSON() layer.
   * Style the lines with a color and weight that will make it stand out on all maps.
   * Add the tectonic layer group variable you created in Step 1 to the map, i.e., .addTo(tectonicPlates) and close the geoJSON() layer.
   * Next, add the tectonic layer group variable to the map, i.e, tectonicPlates.addTo(map).
   * Finally, close the d3.json() callback.
8. Start your Python server and launch the index.html file and confirm that your map with the earthquake and tectonic plate data is similar to the following image.

Your final map should look similar to the following image: