**12-21 notes**

Data Visualization with Leaflet

Goals:

* Explain the benefits of visualizing data with maps.
* Create maps and plot data with the Leaflet.js library.
* Parse data from the GeoJSON format to create map-based data visualizations.
* Describe the concepts of layers and layer controls and how they are applied to add interactivity to maps.

Stu Event Final (Activity in module 14.3) will be useful for Module 14 challenge.

To borrow from leaflet- Style Sheet and 7:14pm in the recording

* Standard piece of code when you’re using leaflet
* <!-- Leaflet CSS --> <link rel="stylesheet" href="https://unpkg.com/leaflet@1.9.4/dist/leaflet.css" integrity="sha256-p4NxAoJBhIIN+hmNHrzRCf9tD/miZyoHS5obTRR9BMY=" crossorigin="" /> <!-- Leaflet JavaScript code --> <script src="https://unpkg.com/leaflet@1.9.4/dist/leaflet.js" integrity="sha256-20nQCchB9co0qIjJZRGuk2/Z9VM+kNiyxNV1lvTlZBo=" crossorigin=""></script>

Add a tile layer to Logic.js file:

// Add a tile layer.

L.tileLayer('https://{s}.tile.openstreetmap.org/{z}/{x}/{y}.png', {

    attribution: '&copy; <a href="https://www.openstreetmap.org/copyright">OpenStreetMap</a> contributors'

}).addTo(myMap);

**Basically, it’s all about Layers.**

**Stu City Markers Solution:**

// Create a map object.

let myMap = L.map("map", {

  center: [37.0902, -95.7129],

  zoom: 5

});

// Add a tile layer.

L.tileLayer('https://{s}.tile.openstreetmap.org/{z}/{x}/{y}.png', {

    attribution: '&copy; <a href="https://www.openstreetmap.org/copyright">OpenStreetMap</a> contributors'

}).addTo(myMap);

// City markers

^ Tile layer line URL has *zxy* not *zyx*

// Add code to create a marker for each of the following cities and to add it to the map.

let cities = [{

  location: [40.7128, -74.0060],

  name: "NYC",

  population: 8550405

},

{

  location: [41.8781, -87.6298],

  name: "Chicago",

  population: 2296224

}];

for (let i=0; i < cities.length; i++) {

  let city = cities[i];

  L.marker(city.location)

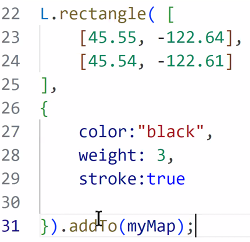
  .bindPopup(`<h1>${city.name}</h1><hr><h3>Population ${city.population.toLocaleString()}></h3></hr>`)

  .addTo(myMap);

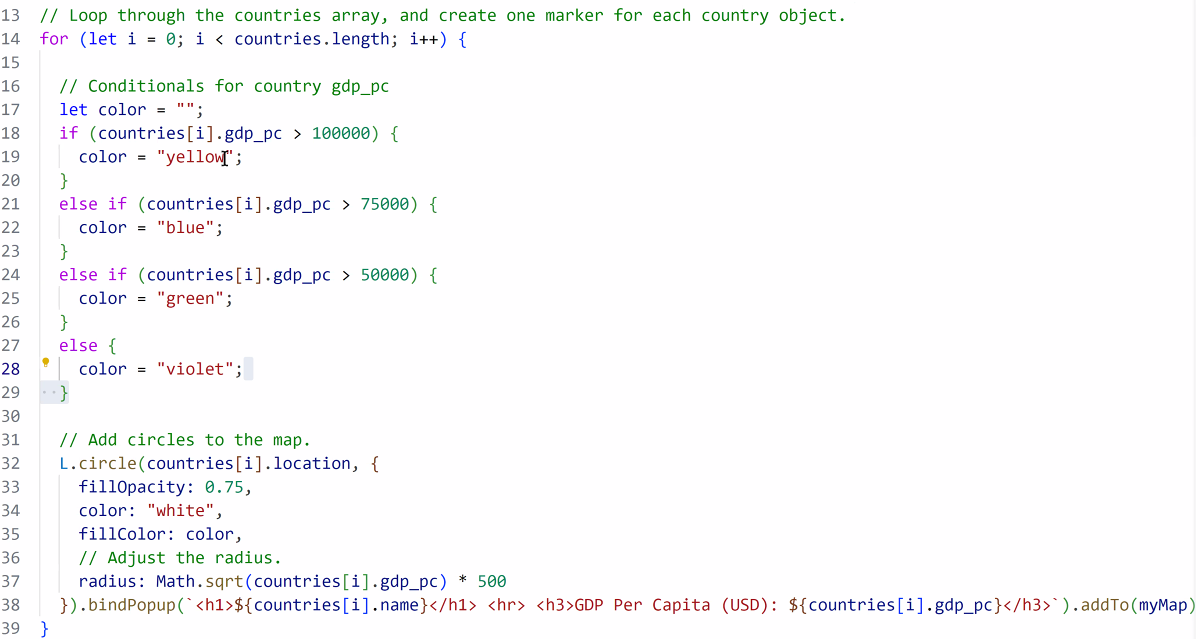
}

Ins Other Markers Solution:

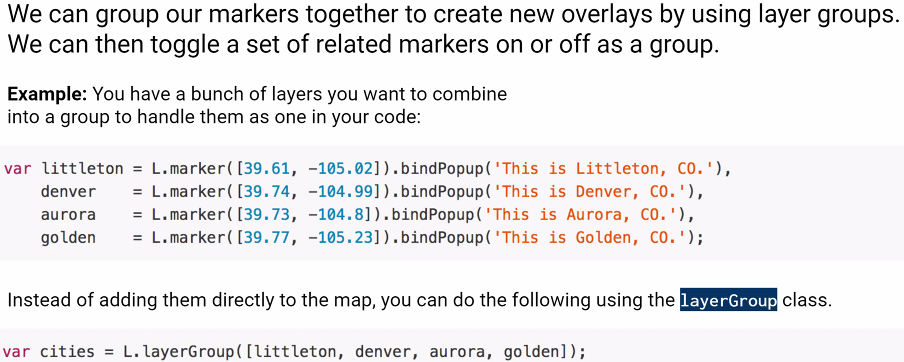




**Stu country GDP Solution:**



You can group layers together.



Ie: You can switch between topographic map and (normal country lines/streetview) maps, you can click a radio button to toggle markers on or off.