* Leaflet Demo
  + Create a simple map
    - Create an HTML document
    - Include the leaflet library
      * CSS

//cdn.leafletjs.com/leaflet-0.7.3/leaflet.css

* + - * JavaScript

//cdn.leafletjs.com/leaflet-0.7.3/leaflet.js

* + - Create a “div” for the map

<div id="map"></div>

* + - Style the new div

body,

html {

height: 100%;

}

#map {

width: 100%;

height: 100%;

}

* + - Create a map with leaflet and assign it to the new div

var map = L.map('map').setView([39.75, -105], 15);

* + - Create a tile layer and assign it to the map.

L.tileLayer('https://{s}.tiles.mapbox.com/v3/{id}/{z}/{x}/{y}.png', {

maxZoom: 18,

attribution: "OpenStreetMap, Mapbox, and NPS",

id: 'nps.k7chg4fe'

}).addTo(map);

* + - Add a marker to the map

var marker = L.marker([39.75, -105]).addTo(map);

* + Modify the simple map to use GeoJSON
    - TODO: We need to decide on a library for AJAX
      * Add reqwest

//cdnjs.cloudflare.com/ajax/libs/reqwest/1.1.5/reqwest.min.js

* + - Load the GeoJSON from CartoDB and from Github

var geojsonLink = 'https://npmap-demo-account.cartodb.com/api/v1/sql?q=select+\*+from+restaurants\_modified&format=geojson';

* + - Add the GeoJSON to the map

reqwest(geojsonLink, function(result) {

L.geoJson(result).addTo(map);

});

* + Style the data
    - Add a custom marker

{

pointToLayer: function(feature, latlng) {

return L.circleMarker(latlng);

}

}

* + - Style the marker

{

color: 'white',

fillOpacity: 1,

radius: 6

}

* + - Define the colors

var colors = {

'Bar': '#8e44ad',

'Cafe': '#e74c3c',

'Pub ': '#16a085',

'Restaurant': '#f1c40f'

};

* + - Link the colors

fillColor: colors[feature.properties.amenity],

* + Add popups
    - Include Handlebars

//cdnjs.cloudflare.com/ajax/libs/handlebars.js/2.0.0/handlebars.min.js

* + - * Explain Handlebars and templating
        + “Handlebars.js is a logicless templating languages that keep the view and the code separated like we all know they should be.”
    - Create a function that generate the popup content

var popupContent = function(properties) {

var content = '<div class="popup">' +

'<span class="title">{{name}}</span><br/>' +

'{{address}}<br/>' +

'<hr/>' +

'<span class="title">Amenity</span>: {{amenity}}<br/>' +

'<span class="title">Cuisine</span>: {{cuisine}}<br/>' +

'<span class="title">Rating</span>: {{rating}}' +

'</div>';

return Handlebars.compile(content)(properties);

};

* + - Add an “onEachFeature” Section
      * Explain how this assigns a popup to each individual point on the map

onEachFeature: function(feature, layer) {

return layer.bindPopup(popupContent(feature.properties));

}

* + View your popups!
  + Leaflet Plugins
    - http://leafletjs.com/plugins.html
    - Leaflet Draw
      * Add leaflet draw CSS

//leaflet.github.io/Leaflet.draw/leaflet.draw.css

* + - * Add leaflet draw js

//leaflet.github.io/Leaflet.draw/leaflet.draw.js

* + - * Add some boilerplate leaflet draw code

|  |  |
| --- | --- |
| var drawnItems = new L.FeatureGroup(); |  |
|  | map.addLayer(drawnItems); |
|  |  |
|  | var drawControl = new L.Control.Draw({ |
|  | draw: { |
|  | position: 'topleft', |
|  | polygon: { |
|  | title: 'Draw a sexy polygon!', |
|  | allowIntersection: false, |
|  | drawError: { |
|  | color: '#b00b00', |
|  | timeout: 1000 |
|  | }, |
|  | shapeOptions: { |
|  | color: '#bada55' |
|  | }, |
|  | showArea: true |
|  | }, |
|  | polyline: { |
|  | metric: false |
|  | }, |
|  | circle: { |
|  | shapeOptions: { |
|  | color: '#662d91' |
|  | } |
|  | } |
|  | }, |
|  | edit: { |
|  | featureGroup: drawnItems |
|  | } |
|  | }); |
|  | map.addControl(drawControl); |
|  |  |
|  | map.on('draw:created', function (e) { |
|  | var type = e.layerType, |
|  | layer = e.layer; |
|  |  |
|  | if (type === 'marker') { |
|  | layer.bindPopup('A popup!'); |
|  | } |
|  |  |
|  | drawnItems.addLayer(layer); |
|  | }); |

* + Push the code to github