

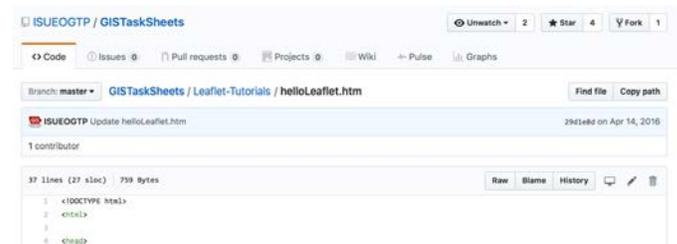
Mapping API's: Leaflet - Layer Control

Welcome to the Essential ArcGIS Task Sheet Series. This series supplements the Iowa State University Geospatial Technology Training Program short course series. The task sheets are designed to provide quick, easy instructions for performing mapping tasks.

The code for this task sheet (**layerControlLeaflet.html**) and previous task sheets can be found on the ISU Geospatial Technology Program GitHub page at <https://github.com/ISUEOGTP/GISTaskSheets>. This task sheet demonstrates how to manage map layers with Leaflet to allow users to toggle map data on and off and switch between basemaps. You will be building a fictitious weather alert map that provides a weather radar layer, sites of interest and alerts.

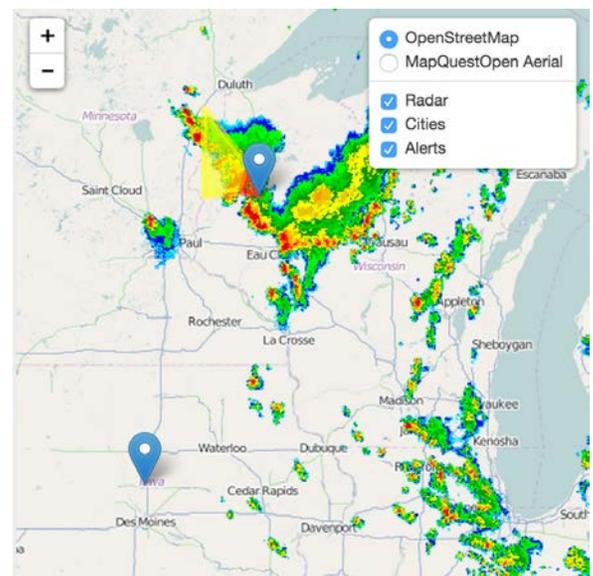
1. Introduction

- First, start with a basic leaflet map setup. Reference the task sheet: *Mapping API's: Leaflet - Getting Started PM2082-14r* to learn how to get this set up, or get the starter code from our GitHub page at <https://github.com/ISUEOGTP/GISTaskSheets/blob/master/Leaflet-Tutorials/helloLeaflet.html>.



2. Add Basemap Layers

- Start by using the basic leaflet setup to display a map and adjust the zoom and map center as desired.
- This map will have two basemap options that the user can toggle between. The first is an OpenStreetMap layer, this layer is already provided in the start up code. Add **var OSM = L.tileLayer()**. Each basemap layer needs a variable name in order for it to be differentiated from and used by Leaflet's layer control feature.
- Next, set **var MapQuestOpen_Aerial** equal to the Map Quest tiles URL, attribution, and subdomains shown to the right.



3. Add WMS Radar Layer

- The radar layer is a web map service (WMS) and is produced by the Iowa Environmental Mesonet (IEM) at Iowa State University. To add this layer, set **var nexrad = new L.tileLayer.wms** and include properties shown below.

```
var nexrad = new
L.tileLayer.wms("http://mesonet.agron.iastate.edu
/cgi-bin/wms/nexrad/n0r.cgi", {
  layers: 'nexrad-n0r',
  format: 'image/png',
  transparent: true,
  attribution: "Weather data &copy; 2015 IEM
Nexrad"
});
```

```
//Base Layers
var OSM =
L.tileLayer('http://{s}.tile.osm.org/{z}/{x}/
{y}.png', {
  attribution: '&copy; <a href="http://osm.
org/copyright">OpenStreetMap</a> contributors',
  maxZoom: 18
});

var MapQuestOpen_Aerial =
L.tileLayer('http://oatile{s}.mqcdn.com/
tiles/1.0.0/sat/{z}/{x}/{y}.jpg', {
  attribution: 'Tiles Courtesy of <a
href="http://www.mapquest.com/">MapQuest</
a>&mdash; Portions Courtesy NASA/JPL-Caltech
and U.S. Dept. of Agriculture, Farm Service
Agency' ,
  subdomains: '1234'
});
```

4. Add City Point Markers

- Identify two cities of interest by adding a standard leaflet marker in the location of the city. Use the **.bindPopup()** property to display the name of the city in a pop-up window when the marker is clicked. *Note: make sure to define each city with a unique variable name as shown below.*

```
//-----Cities
var riceLake = L.marker([45.498333,
-91.738889]).bindPopup('Rice Lake, WI. ');
var ames = L.marker([42.034722,
-93.62]).bindPopup('Ames, IA');
```

5. Add Alert Areas

- Identify two areas of high alert to display on the weather map. Define each alert area with a unique variable name, coordinates for the polygon, and options for outline color, fill color, and fill opacity.

```
//Alerts
var alert1 = L.polygon([
  [45.825833, -91.891111],
  [45.535833, -92.023333],
  [45.498333, -92.638889],
  [46.498333, -92.638889]
],{
  color: 'yellow',
  fillColor: 'yellow',
  fillOpacity: 0.6
});

var alert2 = L.polygon([
  [45.825833, -91.891111],
  [45.498333, -91.738889],
  [45.535833, -92.023333]
],{
  color: 'red',
  fillColor: 'red',
  fillOpacity: 0.5
});
```

6. Layer Group

Layers can be grouped together to allow the user to toggle on and off all cities or all alerts by creating a **L.layerGroup()** that contains an array of features.

```
//feature groups
var cities = L.layerGroup([riceLake, ames]);

var alerts = L.layerGroup([alert1, alert1]);
```

Contact:

Bailey Hanson bahanson@iastate.edu, 515-520-1436 or Professor Christopher J. Seeger, ASLA, GISP cjseeger@iastate.edu, 515-509-0651 for more information about the Geospatial Technology Program. This task sheet and more are available at www.extension.iastate.edu/communities/gis

Iowa State University Extension and Outreach does not discriminate on the basis of age, disability, ethnicity, gender identity, genetic information, marital status, national origin, pregnancy, race, religion, sex, sexual orientation, socioeconomic status, or status as a U.S. veteran. (Not all prohibited bases apply to all programs.) Inquiries regarding non-discrimination policies may be directed to Ross Wilburn, Diversity Officer, 2150 Beardshear Hall, 515 Morrill Road, Ames, Iowa 50011, 515-294-1482, wilburn@iastate.edu.

7. Create the Map

- After the layers and layer groups are created, the map can be generated. Replace the map generator from the start up code with the code to the right. The **layers** option includes a list of the layer names that will be turned on by default when the map loads.

```
//create the map
var map = L.map('map', {
  center: new L.LatLng(44, -91),
  zoom: 6,
  layers: [OSM, nexrad, cities,
alerts]
});
```

8. Layer Control

- Layer control is required by leaflet to be able to toggle layers on and off. Layer control splits layers into those that are basemaps and those that are overlays. Only one basemap can be selected for display at a time. Add a variable named **baseMaps** to contain the basemap name to be displayed in the layer control window. This text is a string and should be surrounded by double quotes followed by the basemap variable name that corresponds to it. The same structure is used to define the overlay maps.

```
//layer control
var baseMaps = {
  "OpenStreetMap": OSM,
  "MapQuestOpen Aerial": MapQuestOpen_Aerial
};

var overlayMaps = {
  "Cities": cities,
  "Alerts": alerts,
  "Radar": nexrad
};
```

- Once the base and overlay layers are defined, **L.control.layers()** is used to add them to Leaflets built in layer control.

```
L.control.layers(baseMaps, overlayMaps).
addTo(map);
```