**GEOG 479 Geospatial Visualization and Visual Analytics**

**Leaflet Basics 2: Vector Layers**

**Leaflet Tutorial Part-Two**

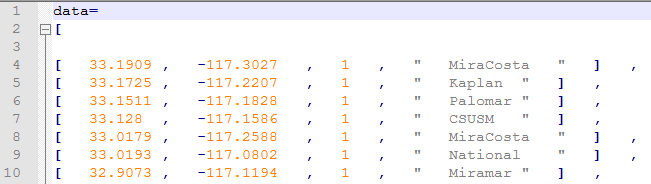
**Data for this lab**

First, you need to download “[**Leaflet\_basics2**](https://drive.google.com/drive/folders/1fwAwPS9xUwheeGRzkCMkYGgUqbIPWv7a?usp=sharing) ” on your computer

Once you download, open up the folder, “**Leaflet\_basics2**” and open up the subfolder, “**exercise**”. Here you have all files and data that you need to complete the exercises. Remember that you will need the entire files under the folder “**Leaflet\_basics2**”. You also need to keep the current folder hierarchy to make the program run.

**1. Visualizing College names in San Diego using popups**

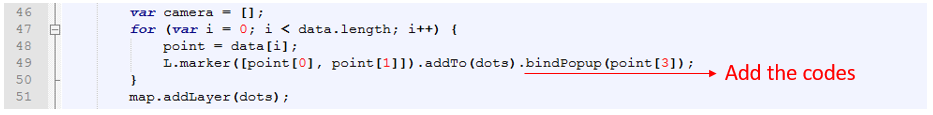
In the exercise last week, you have learned how to visualize multiple points with popup messages. This time, you will learn how to display texts that are connected to a particular point. First, open up college.js. Then, you will see the coordinates of each college in San Diego and the name of the college in the last column.



You are going to display these points on maps and add each popup to show each name of colleges. Run **EX1\_1\_college \_marker.html** and **EX1\_2\_college \_circle.html.** You will see the makers representing the distribution of colleges in San Diego in the first map and the circles in the second map. Our next step is to add popup texts to both marker or point maps.

* 1. **How to add each popup to each marker**

Copy the “**EX1\_1\_college \_marker.html**” and rename the file to “**EX1\_1\_college\_popups\_marker.html”.**

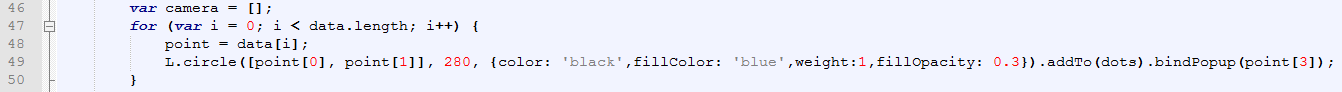


Add the code above in line #49. Point[3] is the **fourth** element (names) of an array called “data” in college.js. “bindPopup” means that you are attaching the popup. After the modification, run this program. Now try to click the marker on your map. Every time you click the maker, you will see a pop up representing the college name.

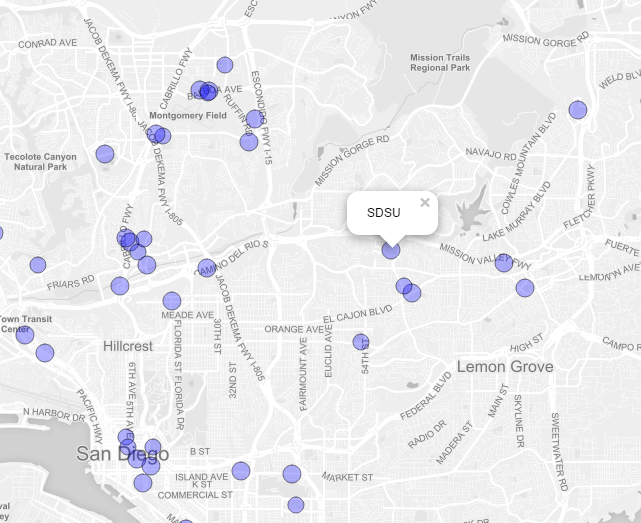


* 1. **How to add each popup to each point**

Second, you will add each popup to each point. Copy the “**EX1\_2\_college\_circle.html**” and rename the file to “**EX1\_2\_college\_popups \_circle.html”.**

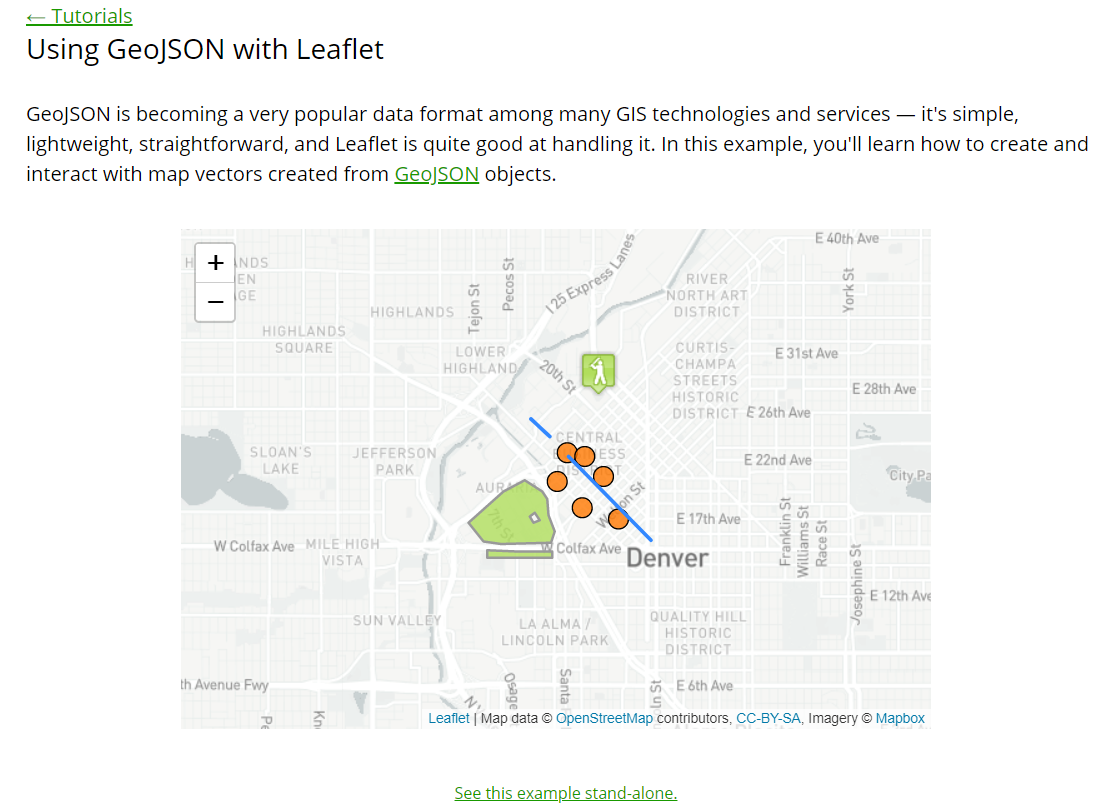


Add the same code “.bindPopup(point[3])” at the end of line #49. After the modification, run this program. Now try to click the marker on your map. Every time you click the maker, you will see a pop up representing the college name. If you have a hard time opening up the pop-up, zoom in, and click the point.



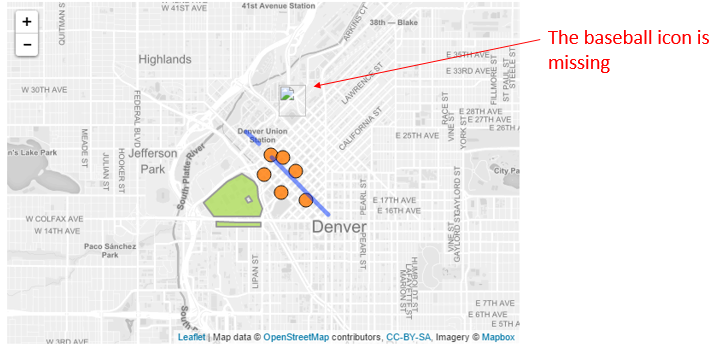
**2. Using GeoJSON with Leaflet.**

On the page of Tutorial(<http://leafletjs.com/examples.html> ), go to “Using GeoJSON with the leaflet”. Then, it should be directed to this website: <https://leafletjs.com/examples/geojson> just like an image below

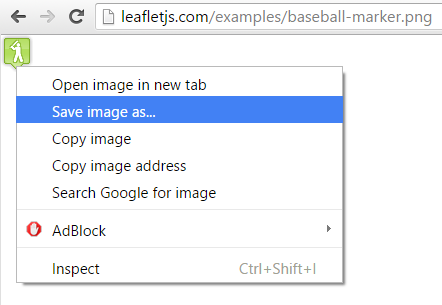


While the Tutorial explains how to code step by step to make the map above, you can also download the entire code. Click “See this example stand-alone” right below the first map. Then, you will see the map on a separate page. On this page, right click your mouse anywhere. You will see the box below. Click “View page source”. Then, you will see the entire code that is needed to draw this map. You will need to save this code in your Notepad++. Copy the entire code and paste it into your Notepad++, save this file and name it **sample-geojson.html.** There is one more file that you need to download, which is **sample-geojson.js**. This file can be downloaded from (https://leafletjs.com/examples/geojson/sample-geojson.js).

Once you save both files (**geojson-example.html** and **sample-geojson.js**) in your folder (you should put both files in the same folder), you can run this code by drag and drop **geojson-example.html** in your browser (Chrome is recommended). Once you run the code, you will see that the baseball icon is missing just like an image below.

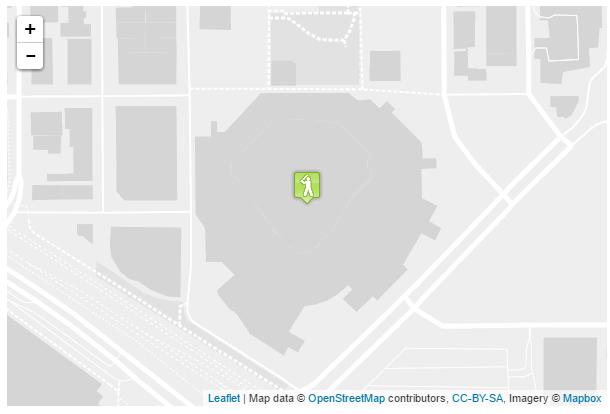


It is because the image is not available in your folder. Download this image from the link <https://leafletjs.com/examples/geojson/baseball-marker.png> To save this image, right click your mouse on the image and click “Save image as…”. The name of this image must be **baseball-marker.png**.



To make the program run, you should put all the files (**geojson-example.html**, **sample-geojson.js,** and **baseball-marker.png**) in the same folder (\leaflet\_basic2\exercise). Read the tutorial step by step to understand how this program (**sample-geojson.html**) works. Once you are done, complete the exercise below.

**Exercise2**. Based on the knowledge that you have learned, write Javascript codes to make a map just like the image below. The map has the baseball icon in the PETCO PARK in San Diego. You can start coding by modifying **geojson-example.html.**



**Tips**

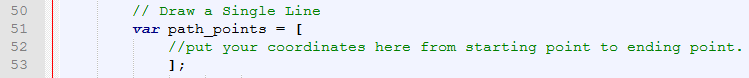
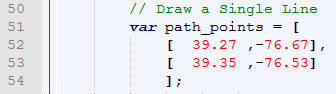
* Step 1. Copy the “**geojson-example.html**” to rename the files to “**EX2\_PETCO.html”** files. Then use the **Notepad++ (editor)** in your computer to edit these HTML files.
* Step 2. Modify the **EX2\_PETCO.html** files based on their elements. For example, change the SetView ([Petco Park coordinates] and the zoom level to 17).
* Step 3. Modify the coordinates in the variable, “coorsField” in the file, “sample-geojson.js
* Step 6. Save the updated “js” and HTML files and test them.

**Tips**: You will be able to get coordinates (Latitude and Longitude) of a point that you want on the website below: http://www.mapcoordinates.net/en Try to click anywhere on the map. You will see a pop up that shows the coordinates of a point that you clicked. You can also use Google Maps directly to find the coordinate of PETCO PARK, San Diego.

**3\_1. Make a single line on the map**

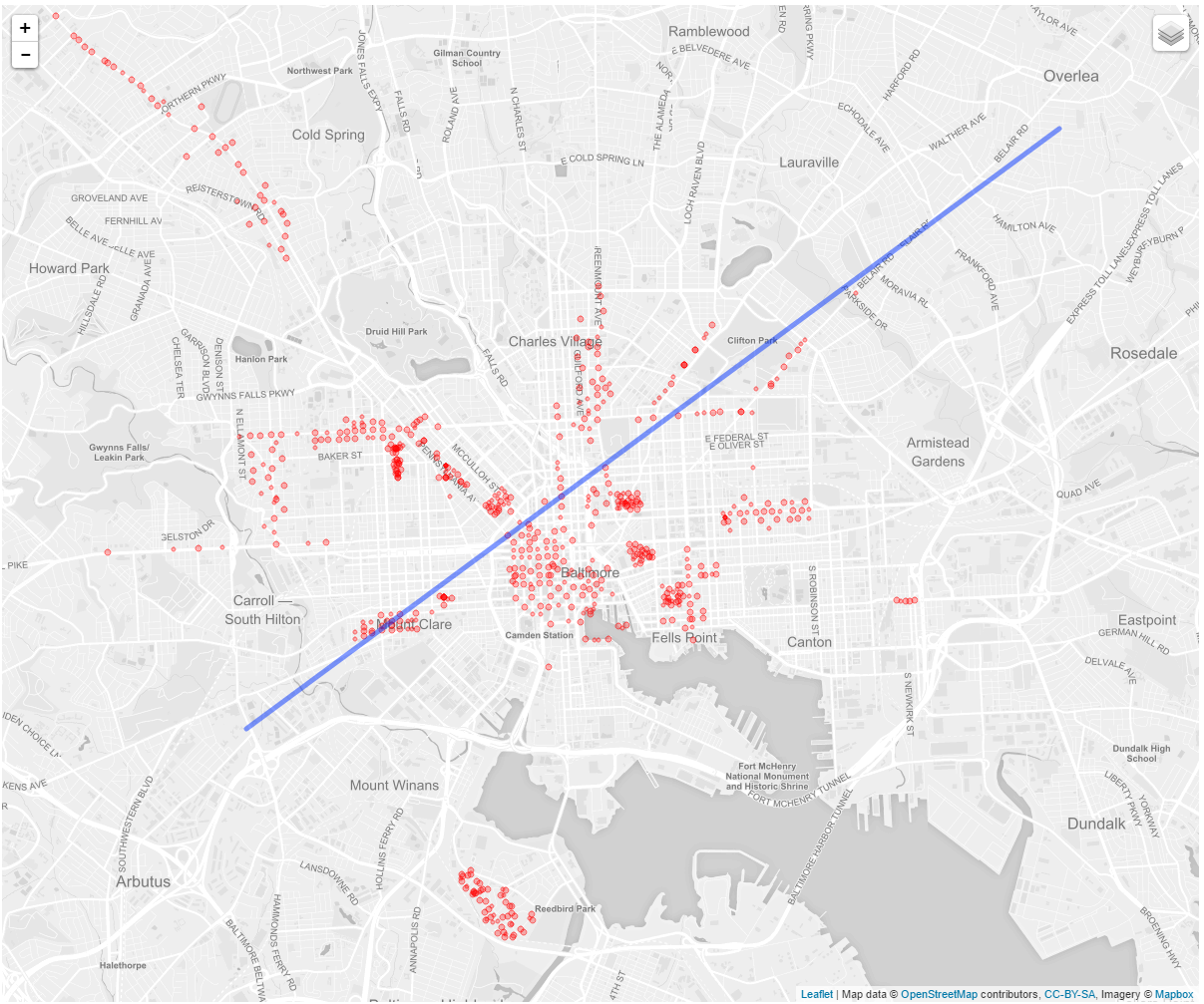
You have learned how to display lines in geojson format in the tutorial above. However, if the lines that you are displaying are simple, you don’t need to use geojson format. In this section, you will learn how to draw simple lines.

Copy “**EX3\_1.html**” and rename the file to “**EX3\_1\_Line\_Single.html”.** If you run the program, you will see the distribution of points. On this map, you are going to draw a line on this map. Open up “**EX3\_1\_Line\_Single.html”** in your notepad++. In line #51, path\_points is an array where you put the coordinates of the beginning point to the endpoint of a line that you want to draw. Modify the line#52, and put the coordinates just like an image on the right below.

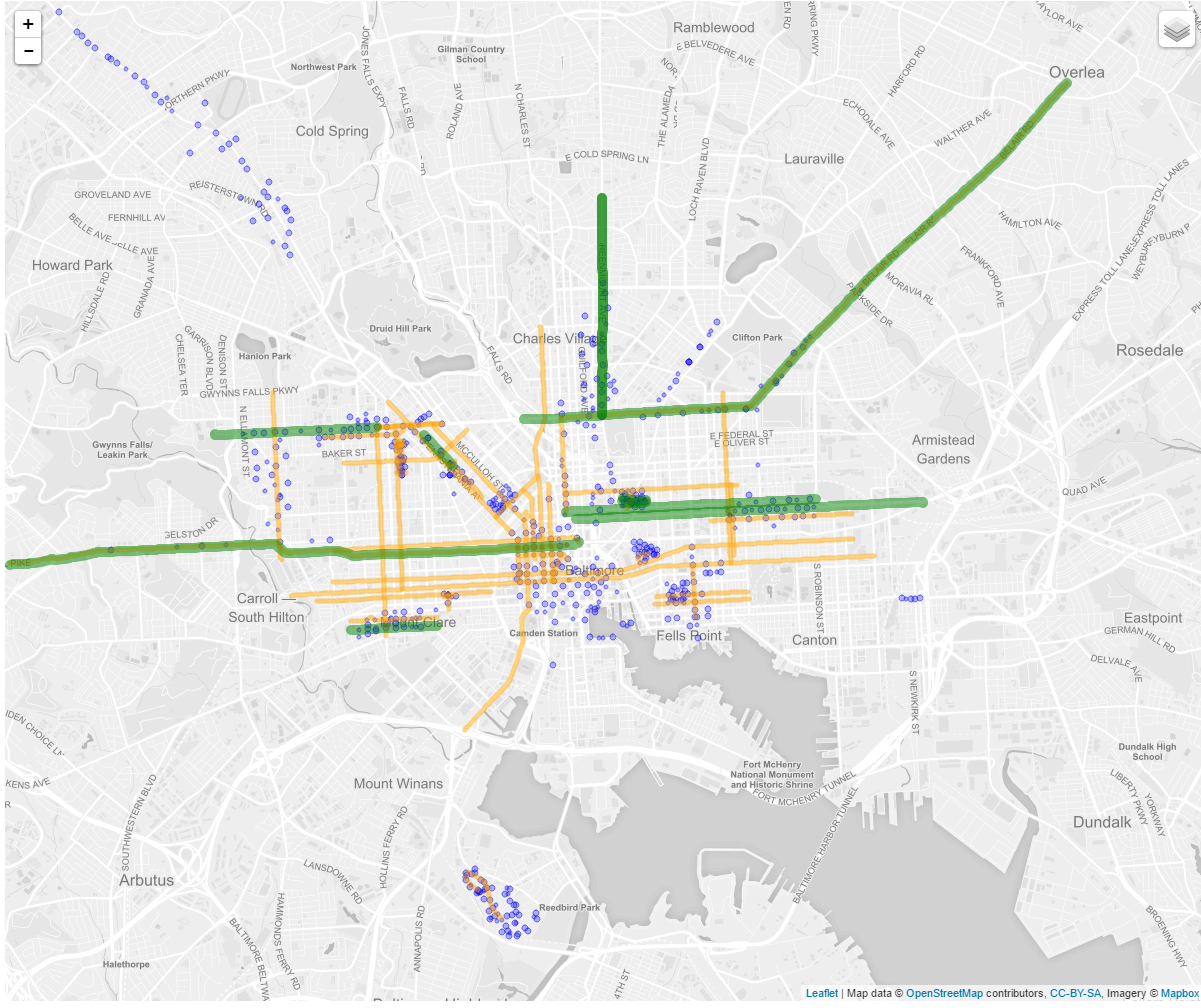
Before modification (image on the left), and after modification (image on the right)

After you complete your modification, run the program **EX3\_1\_Line\_Single.html.** You will see the map below.



**3\_2. Make multiple lines on the map using GeoJSON**.

Now you will learn how to display complicated and multiple lines on the map. At the end of the exercise, you will see the map below. Points are the distribution of surveillance cameras in the city of Baltimore. Orange lines are the roads that have the surveillance cameras within 6 to 9 miles from the road. Green lines are the roads that have the surveillance cameras within 10 to 16 miles from the road.

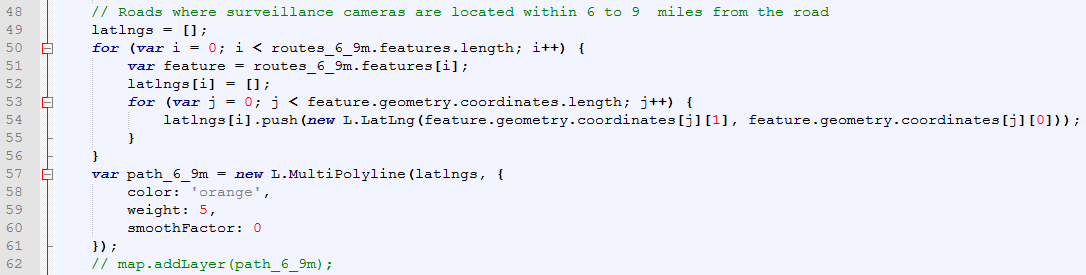


Copy “**EX3\_2.html**” and rename the file to “**EX3\_2\_Line\_Multiple.html”.** If you run the program, you will see the distribution of points. The coordinates of the vertices of lines are already prepared in the **Routes\_6\_9m.js** in GeoJSON format. Please open up these two files and observe how the data look like. In the file, there is a variable, routes\_6\_9m. This variable has all the coordinates of vertices consisting of yellow lines in the above map. This variable is imported in line #19 of the program **EX3\_2\_Line\_Multiple.html**.

The vertices in the variable, routes\_6\_9m, are converted to an object to draw a line from line #49 to the line #61. You can change the color, weight of the line in #58 and #59. Now the object for drawing a line is ready, but it does not display the line on the map yet. The line will be displayed by adding the code below.

map.addLayer(path\_6\_9m);

in the line #62. This code is already there, so you only need to remove the double slash “//” before the code. Then save and run the program, you will see the orange lines on the map.



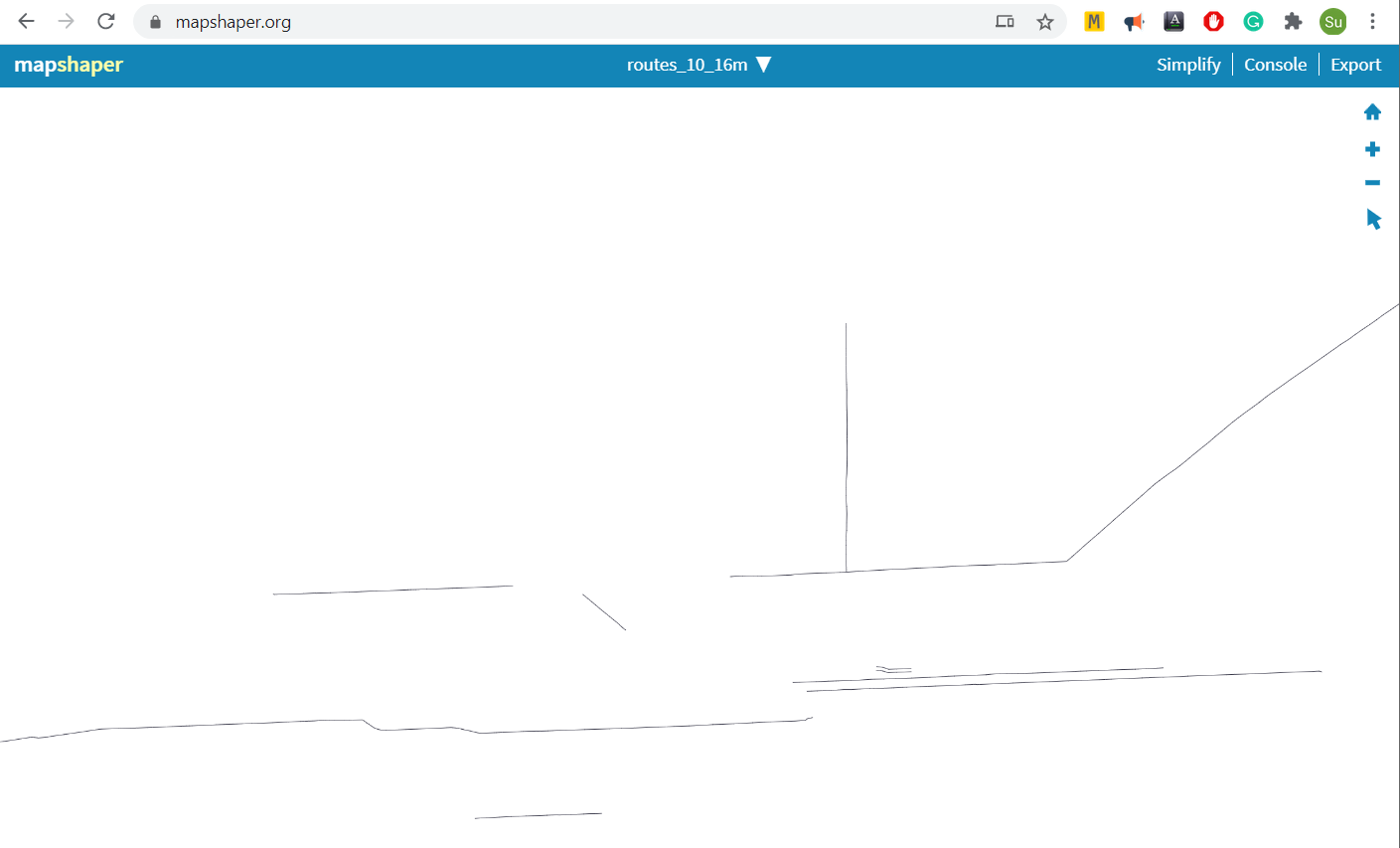


**EX3\_2\_Line\_Multiple.html**

Now you are going to visualize the green lines. This time you will create the GeoJSON object from the shapefile to display the lines. Open up, **Routes\_10\_16m.js**, you will see the file is empty. You are going to create the data from the shapefile (routes\_10\_16m.zip). routes\_10\_16m.zip has a shapefile and other files that are created when you export features from the ArcGIS Desktop. To convert this file to shapefile. Go to the website below.

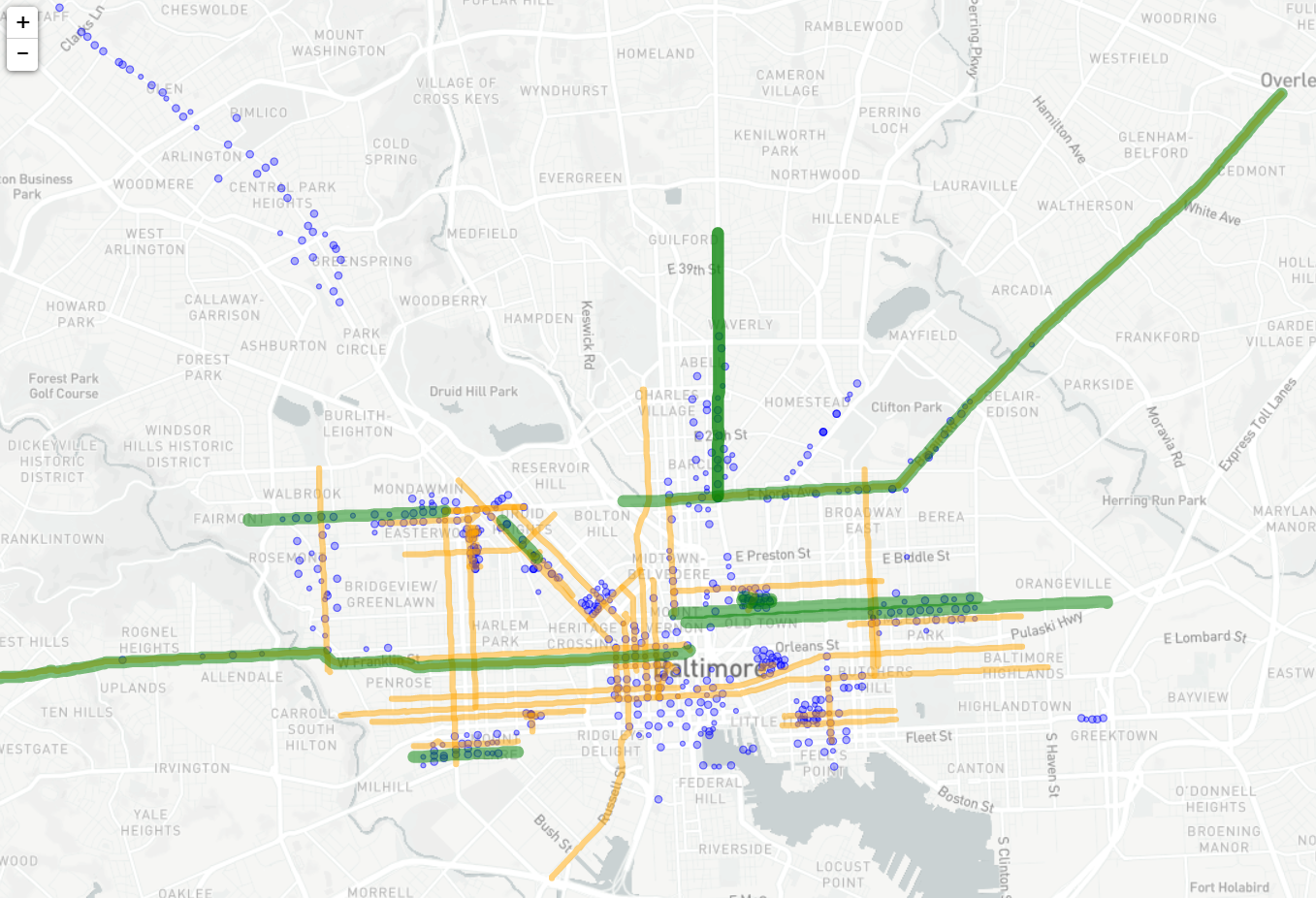
<https://mapshaper.org>

This is the website where you can convert shapefile to geojson. You can upload your data on the first page by drag-and-drop. The file that you have to upload must be zipped. Upload “**routes\_10\_16m.zip**” and click on the button, “**submit**”. Then, you will see the overview of the feature that you uploaded like an image below.

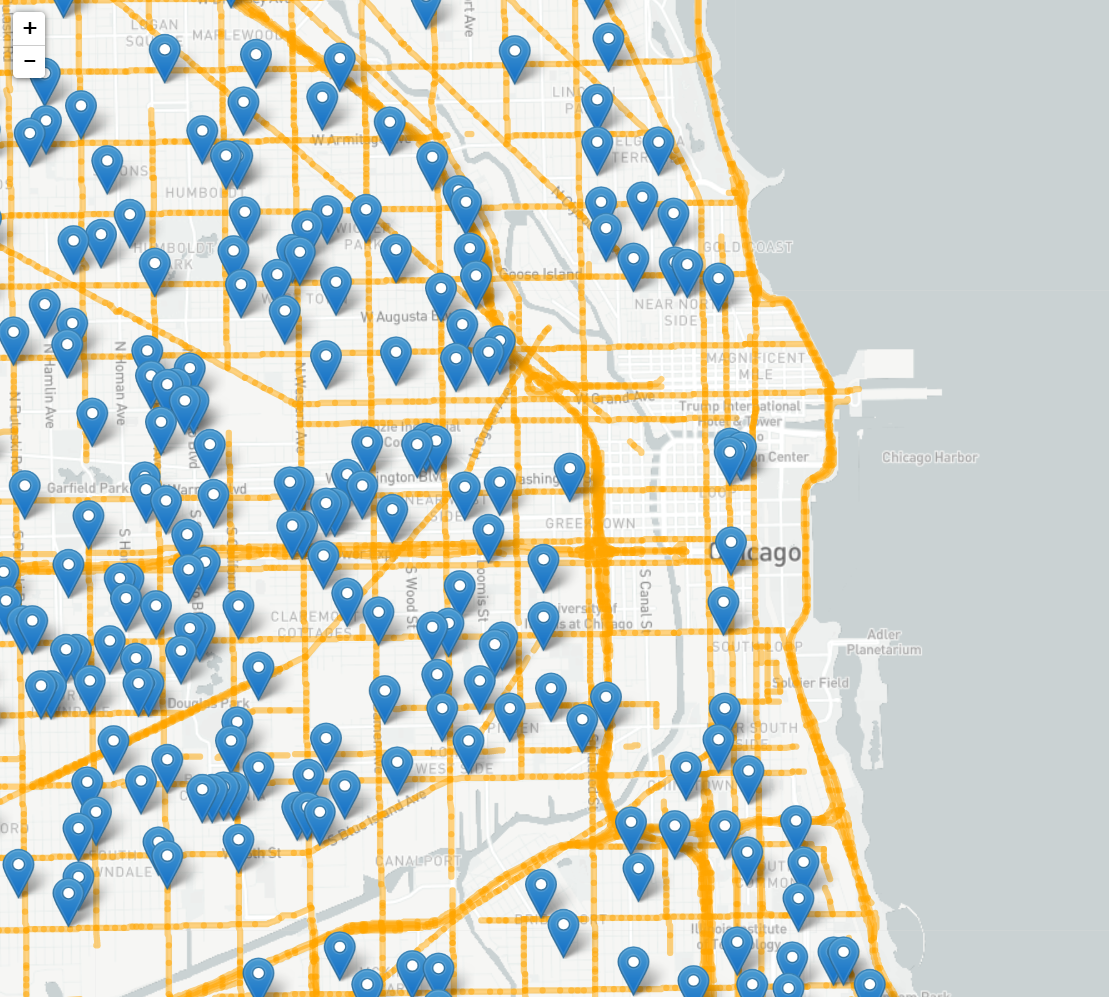


Click Export on the top right corner and export it in GeoJSON. Then, you are going to see **routes\_10\_16m.json** is downloaded. Open up **routes\_10\_16m.js** in your notepad++. Copy the entire data in **routes\_10\_16m.json** and paste it in the second line of **Routes\_10\_16m.js**. **Save** the file.

Next, open up **EX3\_2\_Line\_Multiple.html** in Notepad++.The vertexes in the variable, routes\_10\_16m in Routes\_10\_16m.js, are converted to an object to draw a line from line #67 to the line #79. Remove the double slash in **line 81**, save and run the program. Then, you will see the lines on the map just like an image below.



**Exercise 4**. Based on the knowledge that you have learned, write JavaScript codes to make a map just like the image below. The map shows the distribution of public schools in Cook County in Chicago. and also shows the major streets in Chicago.



**Tips**

* Step 1. Copy the “**EX4\_college\_name.html**” and rename the files to “**EX4\_school\_name\_streets.html”** files. Then, use the **Notepad++ (editor)** in your computer to edit these HTML files.
* “**EX4\_college\_name.html**” contains the location of colleges and freeways in San Diego , so you need to change the map center and zoom level to the Chicago area.
* A major street in Chicago is available in Major\_Streets.js. Examine the geojson data carefully.
* public schools in Chicago are available in school.js
* Delete double slash in the line #68 in **EX4\_college\_name.html**.
* Test the new **EX4\_school\_name\_streets.html**

**Submission Instructions: You will submit everything in your google drive.**

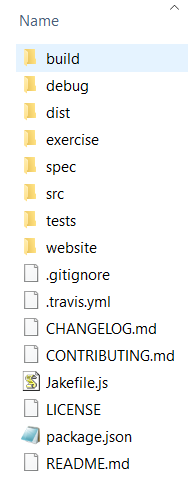
**What to turn in (due day: 04/09/2021 until 11:59pm)**

1. **Submit the following HTML codes.**

* **EX1\_1\_college\_popups\_marker.html**
* **EX1\_2\_college\_popups \_circle.html**
* **EX2\_PETCO.html**
* **EX3\_1\_Line\_Single.html**
* **EX3\_2\_Line\_Multiple.html**

**EX4\_school\_name\_streets.html**

You should test each of the programs and make sure that they work properly. In addition to the HTML files, make sure to include all files that you need to make your HTML file run. All files should be included in one folder. Your folder name should be Lab3\_firstname\_lastname **in the google drive that I share with you**. For example, make a folder such as “Lab3\_Su\_Han”. In this, make a folder, “exercise” and put the HTML files listed above. You will have to submit folders and files like below:



If your HTML file does not run because of missing libraries, files, or data, you are not going to get any credit. So please make sure to submit the runnable code. Do not submit unnecessary files.

1. **Write a short essay to describe the following issue. What is the difference between shapefile and GeoJSON?**

In the end, you will have a folder named Lab1\_firstname\_last name and in the “exercise” folder you will have HTML files and a word file below.

* Lab3\_firstname\_lastname.docx
* EX1\_1\_college\_popups\_marker.html
* EX1\_2\_college\_popups \_circle.html
* EX2\_PETCO.html
* EX3\_1\_Line\_Single.html
* EX3\_2\_Line\_Multiple.html
* EX4\_school\_name\_streets.html

After you submit your lab, do not make any changes. The modified date of your files or folders will be considered as the submission date.