

Getting Started with Mapbox

Link to form: <https://forms.gle/5AM3KThS172x49MUA>

Learning Objectives

- Students will use Python to prepare a dataset for display in Mapbox.
- Students will create a geojson using a webtool for the Mapbox environment.
- Students will create a graduated symbol map based on EPA Toxic Release Inventory Data.
- Students will create a public facing web page for displaying their Mapbox map.

Introduction

One of the most popular methods of creating web maps is the use of JavaScript. JavaScript is a programming language for adding to web page interaction and functionality when paired with HTML and CSS. The Mapbox company has built a web mapping system around JavaScript that allows users to create beautiful interactive maps. This lab is going to introduce you to the Mapbox suite of tools and the very basics of Mapbox GL, the JavaScript library that allows for expansion of map functionality.

A great place to get started with Mapbox learning is [this tutorial](#).

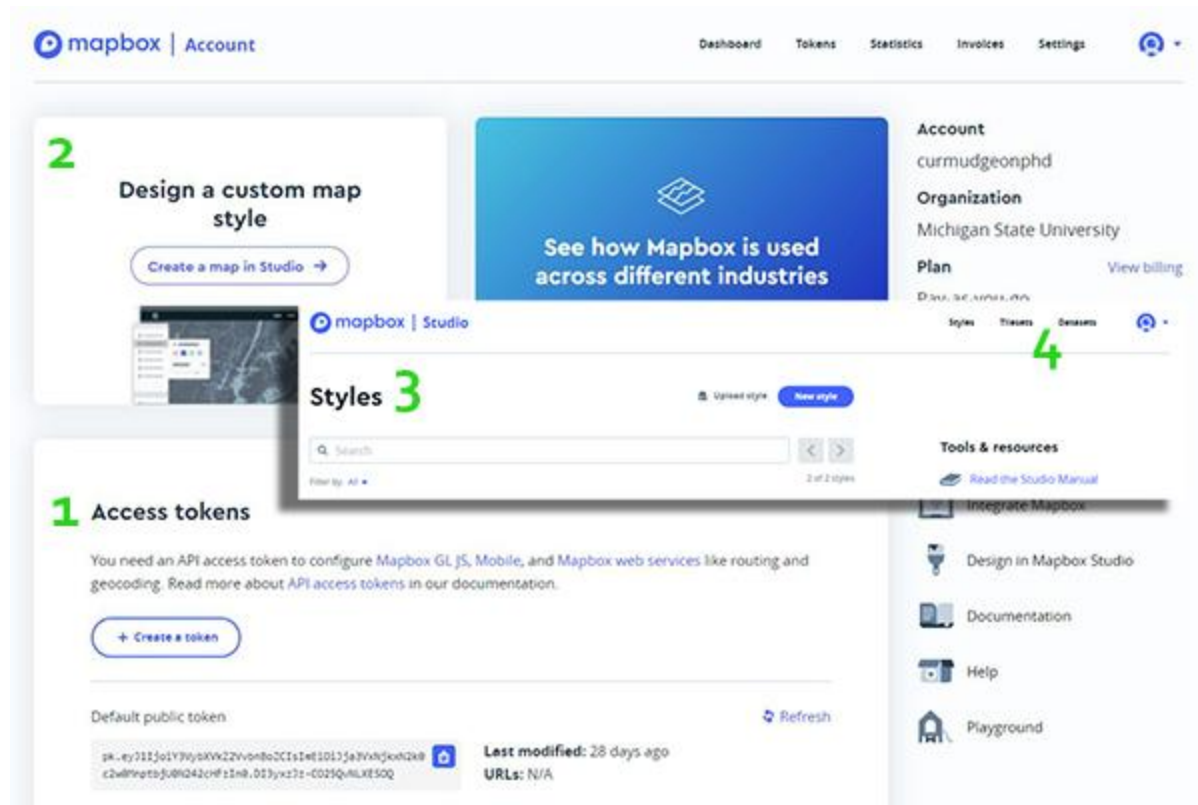
To obtain the files necessary to complete this lab see the github repository here:

[https://github.com/raechelportelli/raechelportelli.github.io/tree/master/cart1/Labs/Lab 10](https://github.com/raechelportelli/raechelportelli.github.io/tree/master/cart1/Labs/Lab%2010)

Signing up for Mapbox

If you do not already have a Mapbox account, you will need one for this lab. You can get a free account at [Mapbox.com](https://mapbox.com). I suggest using your school email for this because I believe you get more perks if you are a student or associated with a university.

Once you've signed up and logged into your account you will see several different tools available. Let's walk through them.



1. Access Tokens

Your access token is the link between your account map designs and your webpage. You will use this link in your HTML to let the web browser know where your map styles are located and that you have the right to access them.

2. Design a custom map style

The main workhorse of mapbox is Mapbox Studio. You can access this tool through this link. The Mapbox Studio is a Graphical User Interface that allows easy manipulation of the design elements for the web map.

3. Styles Page

When you click on the *Design a custom map style* you will be taken to the Styles page. It's here that you can create your own custom style using the *New Style button*. This takes you to the Mapbox Studio and fully functional GUI.

4. Tilesets and Datasets

You can add your own custom data and tiles to Mapbox. Tiles are a collection of raster or vector data broken up into a uniform grid of square tiles at up to 22 preset zoom

levels. Datasets are geojson files that contain geospatial data and that you can use in your map. Datasets are distinct from tilesets in that datasets can be edited on a feature-by-feature basis, but cannot be used directly in Mapbox Studio style. You would need to export your dataset to a tileset to use it in your style.

Mapbox Studio Basic Design Interface

1. In the Styles page click on new style and select the *Monochrome* style. You will be prompted to pick a color. You can choose whichever color you wish.
2. Once you've chosen a color you will be taken into the Studio interface. On the left side you will see a table of contents similar to what you work with in ArcGIS Pro.

The *Components* tab will be highlighted and you will see a list of components, all of which are geography feature types that make up the base map. For example, the Place labels component.

Click on one of the components and take a look at the various properties that are available. Try changing one of those parameters and notice what it does on the map.

3. Notice on the bottom of the components list there are two more controls. *Color* and *Typography*. These controls change the overall styles of the map. For example, you can change the base color for your monochrome map with the Base tool.
4. In addition to the Components tab on the top you will see a Layers tab. This is where the detailed components are at. For example, on the Components tab you can see Road Networks, and control the overall style of the road networks. With the Layers you can control multiple aspects of the Road network.
5. Create a basemap you think would be appropriate for mapping toxic chemical release data.

Data Preparation in Python

We have seen several libraries for working with spatial data in Python this semester. Geopandas is a straightforward library for manipulating geospatial data. For this portion of the lab you will use geopandas to select a subset of the TRI data to map. I will not go through every step as you have already been instructed in this process multiple times this semester. If you need help, please see the [geopandas web page](#) or [pandas web](#)

[page](#) or [Stack Exchange](#). You can do all of the necessary preprocessing using JupyterLabs online.

1. Go to the [Jupyter Project webpage](#) and select Try Jupyter Lab and select a new Python 3.0 console. A new Jupyter lab notebook tab will appear (Untitled.ipynb).
2. Add the EPA TRI csv to the Jupyter notebook using the *up arrow button* (upload). This CSV file was acquired from the EPA website and information about the data set can be found [here](#).
3. Next add the Jupyter notebook file that I've provided using the upload button. The instructions for this portion of the lab are written there. I have left blank code lines where you should try writing your own lines of code to accomplish the given task. In the end you will have a csv file. Proceed with this csv file in the next step.

Converting CSV to geojson

You could write your own code to convert a csv file to a geojson in Python. However, for this lab we will use the tool available [here](#).

1. Upload your CSV file containing your TRI data subset by a single state.
2. Set the latitude and longitude columns under *Choose Output Options*. Note that they are asking for the column number, not the number in the column name.
3. Finally generate output. The map on this page will populate with your geojson data so you can verify that you have success.
4. Finally, download the geojson file to your computer.

Adding your GeoJSON data to Mapbox Datasets

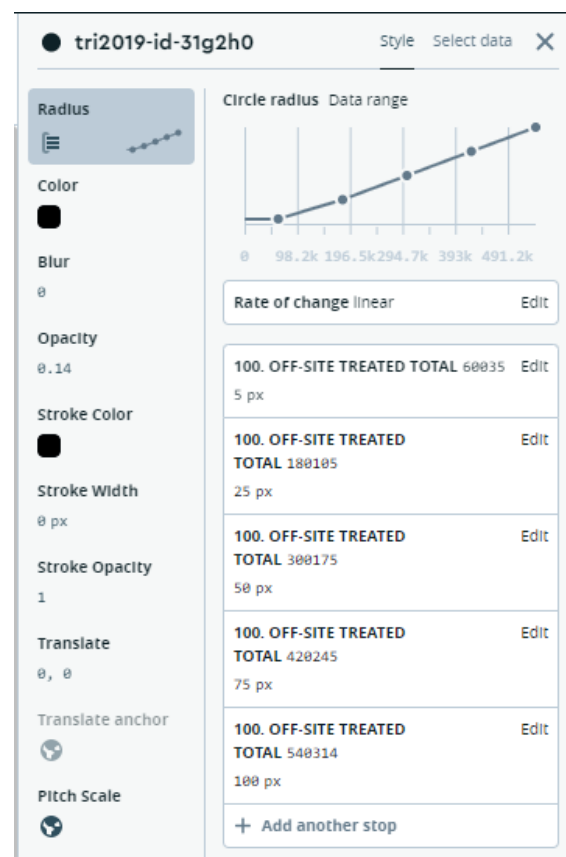
1. In Mapbox, navigate to the Tilesets page (via Styles page).
2. Select *New tileset* and navigate to your geojson file to create a new layer. If you are successful you will see the new tileset populated under the search bar.
3. Click on the tileset in the list. It should open a new page containing a map and your data. You are able to directly edit your own data in this window.
4. Navigate back to the tileset page and copy the tileset name. You will need this in a minute!
5. Next, navigate back to the *Mapbox Studio* User Interface and select your monochrome map style.
6. Highlight the *Layers tab* in Mapbox Studio.

7. To add a new layer, click on the + sign button. This will open a new menu called New Layer.
8. To define the source for your new layer paste the tileset name you copied from step four above.
 - a. NOTE: Your layer will show up in the popup menu that lists available layers; however, if you have a large number of layers it can become difficult to find the one you need.
 - b. NOTE: You could also use the Upload data function to add your geojson in this window.

You should see your data layer added to the list of layers in the Layer menu. It will not show up in your Components tab,

9. Click on your Layer in the list. You will see the menu of style properties pop up. You can change things like color and size for your point symbols directly. You will style your point size based on the TRI OFF-SITE TREATED TOTAL data.

- a. Select *Style across the data range* and select the correct attribute.
- b. To add changes in the map symbol using the data value, similar to the graduated symbol style in ArcGIS Pro, click on Add another stop. Do this until you have five classes like in the picture.
- c. For each of the divisions change the symbol size so the point symbol gets progressively larger. You should see the change reflected in your map. Note in the example here, I've exaggerated the size change quite a bit so you can see it clearly in the graph. Do use your best judgment in your own map!



10. Once you have styled your basemap and your points, you will need to take one last step! Click on the *Publish* button in the upper right-hand corner. This will make your map available for use on your webpage.

- a. NOTE: Make sure you use the Publish button every time you want to update the map design on your webpage. It does not update automatically.

11. DO NOT CLOSE MAPBOX

Displaying your map on a web page

For this last component, you will need to create your own webpage. To do this, I suggest using github. You can find instructions for how to do this [here](#).

One note about creating a github web page, make sure the name of your page is the same as your username. For example, my username is raechelportelli, so my github.io site is raechelportelli.github.io. There is a step in these instructions that instructs you to create a repository and name it username.github.io, if you do not use your current username, it will not work.

12. Open the HTML document I have provided. It is already set up to display your map. You just need to customize the code so that it points to your mapbox map.
 - a. Replace the access token after the variable *mapboxgl.accessToken* and replace it with your own access token.
 - i. Your access token is available on the opening page in Mapbox under Access Tokens (see 1 in figure above)
 - b. Replace the style URL with your own style url.
 - i. Your style url is available on the Styles page. Click on the three dots at the end of your style name (by the upload button). At the bottom of this popup window there is a copy button to capture your personal URL.
 - ii. Replace the coordinates with the center point of the state you are mapping. You can easily find this using Google.

The location of these three changes is highlighted in the figure below.

Written By Raechel Portelli

Updated 11/10/2021

```
<!DOCTYPE html>
<html>
<head>
<meta charset="utf-8">
<title>Display a map on a webpage</title>
<meta name="viewport" content="initial-scale=1,maximum-scale=1,user-scalable=no">
<link href="https://api.mapbox.com/mapbox-gl-js/v2.6.0/mapbox-gl.css" rel="stylesheet">
<script src="https://api.mapbox.com/mapbox-gl-js/v2.6.0/mapbox-gl.js"></script>
<style>
body { margin: 0; padding: 0; }
#map { position: absolute; top: 0; bottom: 0; width: 100%; }
</style>
</head>
<body>
<div id="map"></div>
<script>
  mapboxgl.accessToken = 'pk.eyJ1Ijo1Y3VyYXVkbWVbnBoZCIsImEiOiJja3VxNjktNzZkoc2woMnptbjUoNz42cHFzIno.Dl3yxzJz-CO25QvNLXES0Q';
  const map = new mapboxgl.Map({
    container: 'map', // container ID
    style: 'mapbox://styles/curmudgeonphd/ckvtznp8oogup14p4gcu7voiww', // style URL
    center: [-114.4720, 44.0682], // starting position [lng, lat]
    zoom: 11 // starting zoom
  });
</script>

</body>
</html>
```

13. Save your html file.

14. Upload your html file to your webpage folder.

- a. In the case of github you will upload the file to your username.github.io repository.
 - i. Click on *Add file*, and *Upload files*.
 - b. Make sure the repository is public. To do this click on *Settings* and then *Manage access*.
 - c. Under the heading Who has access should say *Public Repository*. If it does not click on the Manage button and change it to a public repository.
15. Navigate to your webpage. You should be able to see your map on the page now. For example raechelportelli.github.io/map.html looks like this:

