GIS for Designers
VT | A+D | SU22

TUTORIAL 6 | JUMP INTO MAPBOX

Zoom: **Wednesday 06.08, 6:30-7:30pm**
https://virginiatech.zoom.us/j/2981092726

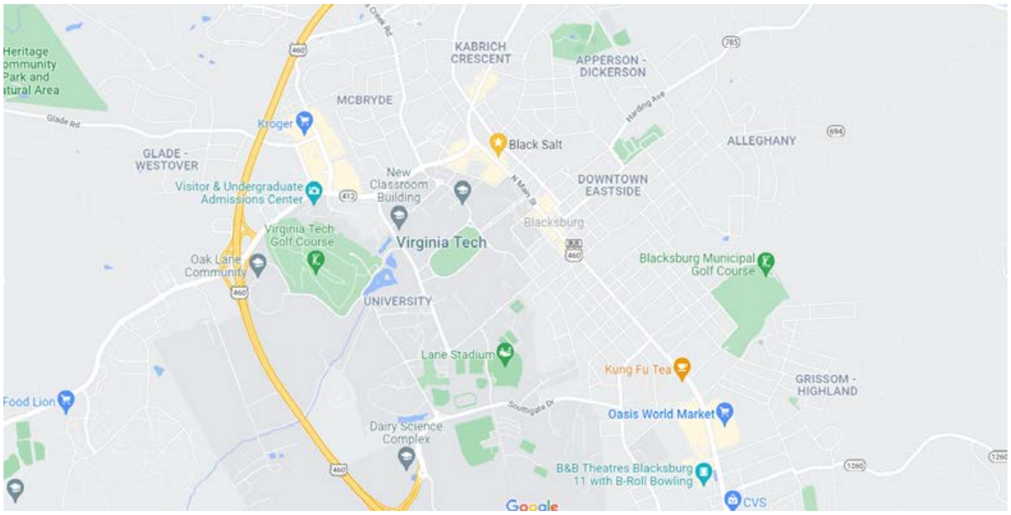
Goals

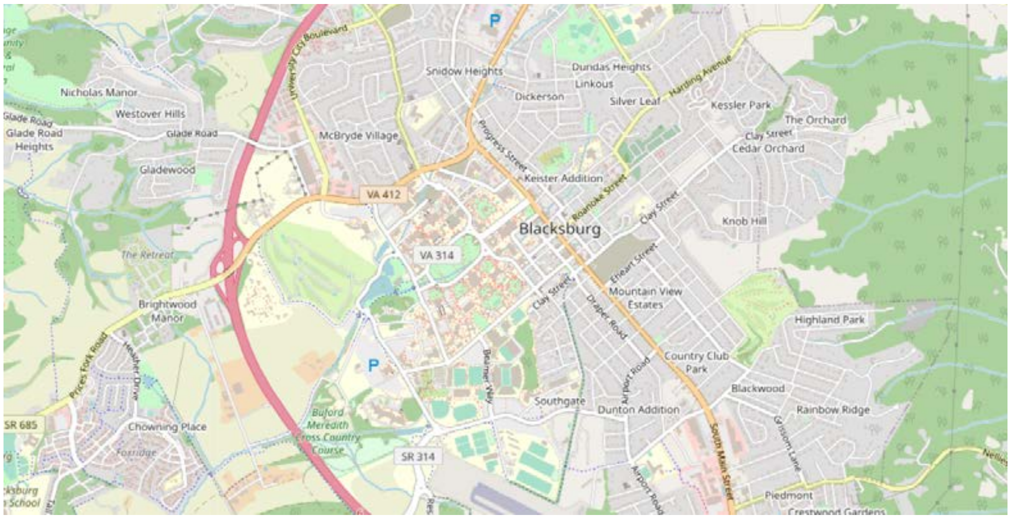
- Gain familiarity with Mapbox platform.
- Understand how to style layers and base map.
- Upload Census shapefile as data layer from QGIS.
- View map in browser.

Intro

Today, you’re going to create a Mapbox base map style and add some basic data.

Think of your basemap as a styled map background. For example, take a look at the difference between Google Maps and Open Street Maps basemaps. Notice the difference in color, labeling, lineweight, information density, and so on. Your basemap is the canvas for information that you want to show. Google Maps emphasizes roads and points of interest – a car-based perspective. OSM, on the other hand, shows a less hierarchical urban fabric of land use as well as roads and places.





Mapbox terms:

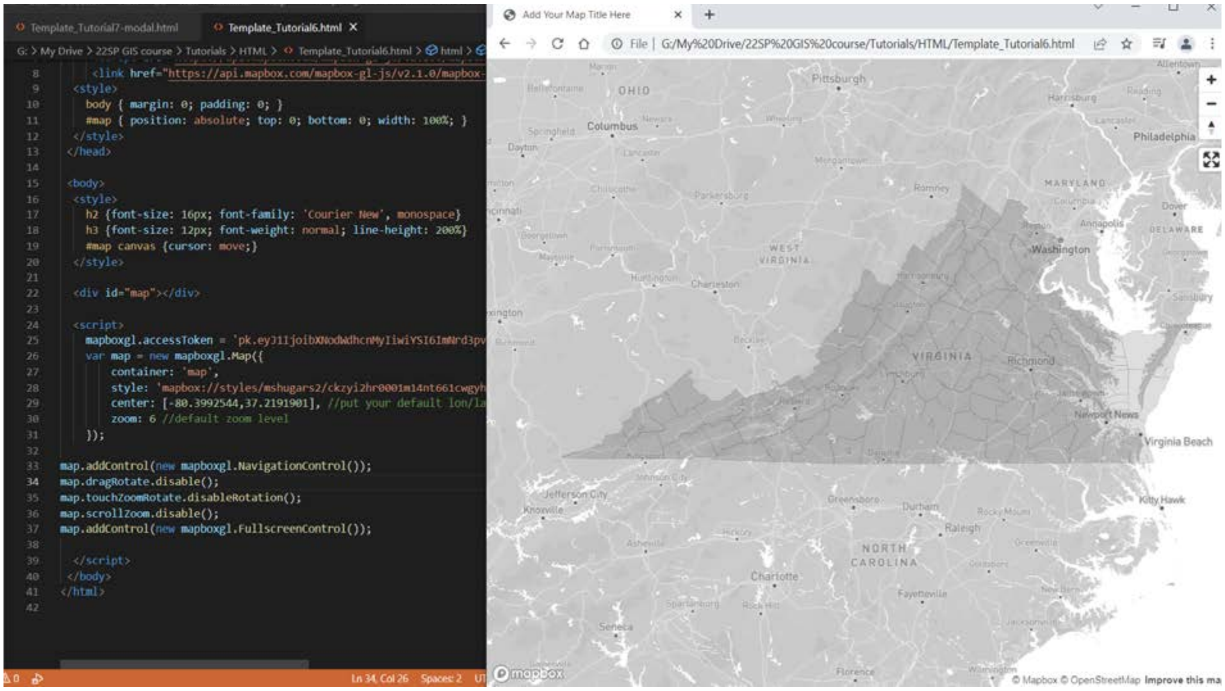
- **Style**: base map
- **Studio**: where you create and edit basemaps
- **Tileset**: map image data that styles by zoom level; generally, basemap information
- **Dataset**: as opposed to tilesets, datasets contain a specific set of data (eg location of community centers, or industrial land use polygons)
- **Access token**: unique id number that you create for your account and use to activate your map in HTML code
- **Component**: groups of shapefile or information layers – generally tilesets – for instance, “land water & sky” or “road network”. These are tilesets made by Mapbox (you can’t upload or make your own component).
- **Layer:** specific subset of data, for instance “water” or “sky” or “bridges”; each component contains multiple layers. When you upload your own shapefile or database information, you can add them as layers to your map.

Visual you can style with Mapbox Studio:

- color and outline of **buildings**, **waterways, roads, natural features, and ground**
- content, size, font, and color of **labels**
- what content displays at which **zoom** levels
- map **cant** (view angle), and **fog**
- note: Mapbox does not show topography lines by default, but you can add the Tileset as a layer to your basemap

You can also **upload** custom **Datasets** or **Tilesets** to your Mapbox base map. This information can be point data, lines, or polygons – that is, traditional Shapefiles. It can also be csv spreadsheets with geographic information in it (latitude and longitude), or even Geo-located raster images (GeoTIFFs). In this Tutorial you’ll upload some useful census shapefile data from QGIS

On **Mapbox** you create the map **object** with all of its relevant visual and informational layers. With **code**, you **interact** (and allow other people to interact) with that object. Once you embed the map object in a website, you will add any interactive features, such as on-hover information popups or clicks, with code.



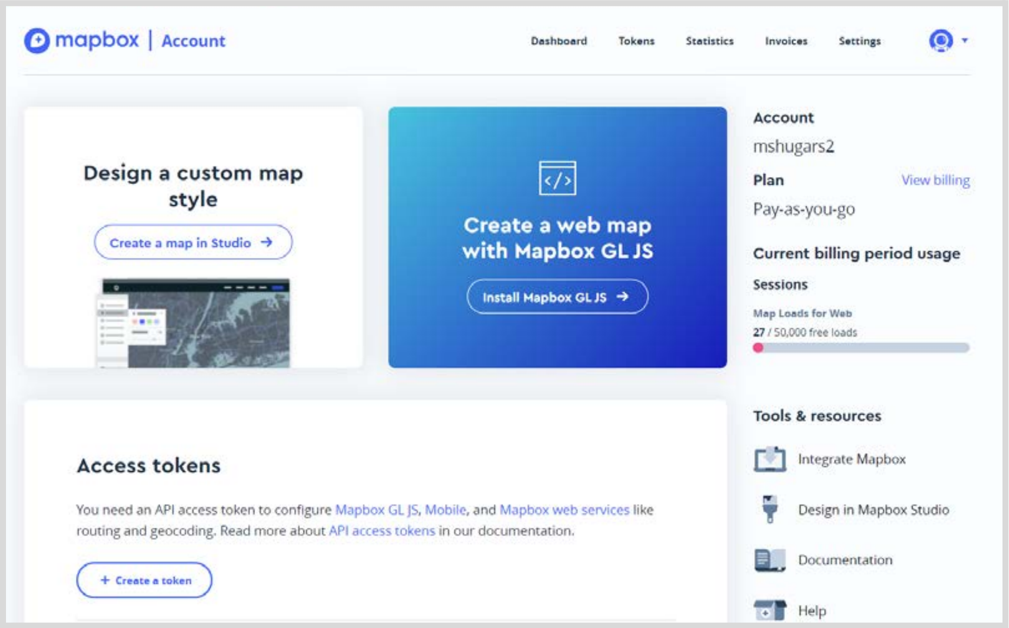
```
      <link href="https://api.mapbox.com/mapbox-gl-js/v2.1.0/mapbox-
    <style>
      body { margin: 0; padding: 0; }
      #map { position: absolute; top: 0; bottom: 0; width: 100%; }
    </style>
  </head>

  <body>
    <style>
      h2 {font-size: 16px; font-family: 'Courier New', monospace}
      h3 {font-size: 12px; font-weight: normal; line-height: 200%}
      #map canvas {cursor: move;}
    </style>

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

    <script>
      mapboxgl.accessToken = 'pk.eyJ1IjoibXNodWdhcnMyIiwiYSI6ImNrd3pv
      var map = new mapboxgl.Map({
          container: 'map',
          style: 'mapbox://styles/mshugars2/ckzyi2hr0001m14nt661cwgyh
          center: [-80.3992544,37.2191901], //put your default lon/la
          zoom: 6 //default zoom level
      });

  map.addControl(new mapboxgl.NavigationControl());
  map.dragRotate.disable();
  map.touchZoomRotate.disableRotation();
  map.scrollZoom.disable();
  map.addControl(new mapboxgl.FullscreenControl());

    </script>
   </body>
  </html>
```

Step 1: Create a Mapbox account.

Mapbox is free for non-commercial use (ie if your map has fewer than 50,000 loads per month).

https://account.mapbox.com/

After confirming your your account, you will see the Mapbox home page:

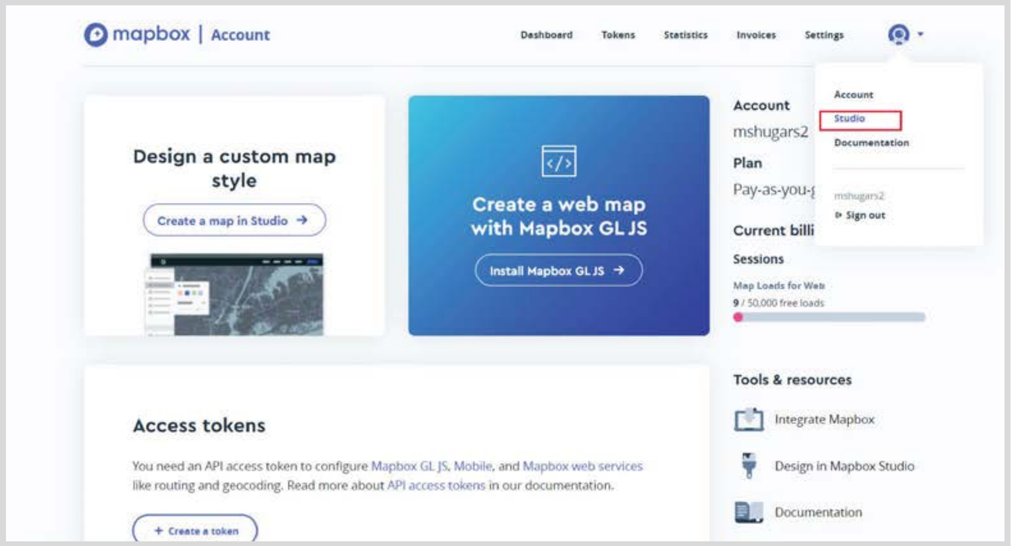


Account
mshugars2
Plan
Pay-as-you-go
Current billing period usage
Sessions
Map Loads for Web
27 / 50,000 free loads

Access tokens

You need an API access token to configure Mapbox GL JS, Mobile, and Mapbox web services like routing and geocoding. Read more about API access tokens in our documentation.

1b Navigate to the Mapbox Studio, where the visual magic happens, by clicking on your profile icon in the upper right and selecting "**Studio**"



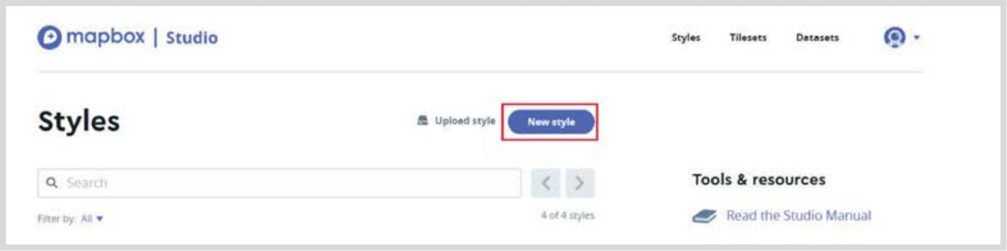
Account
mshugars2
Plan
Pay-as-you-
Current billi
Sessions
Map Loads for Web
9 / 50,000 free loads

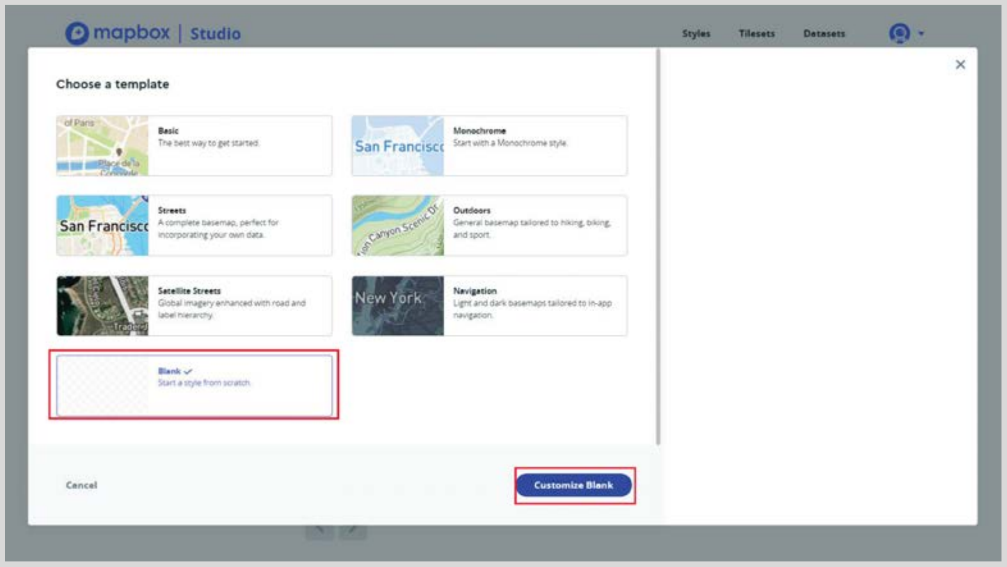
Access tokens

You need an API access token to configure Mapbox GL JS, Mobile, and Mapbox web services like routing and geocoding. Read more about API access tokens in our documentation.

Step 2: In Mapbox Studio, create a new Map Style.

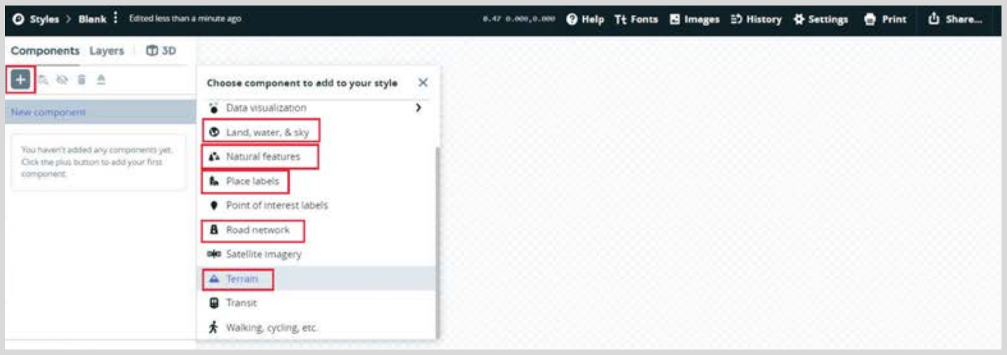
2a Click “**New Style**” and scroll down to **Blank** map style.



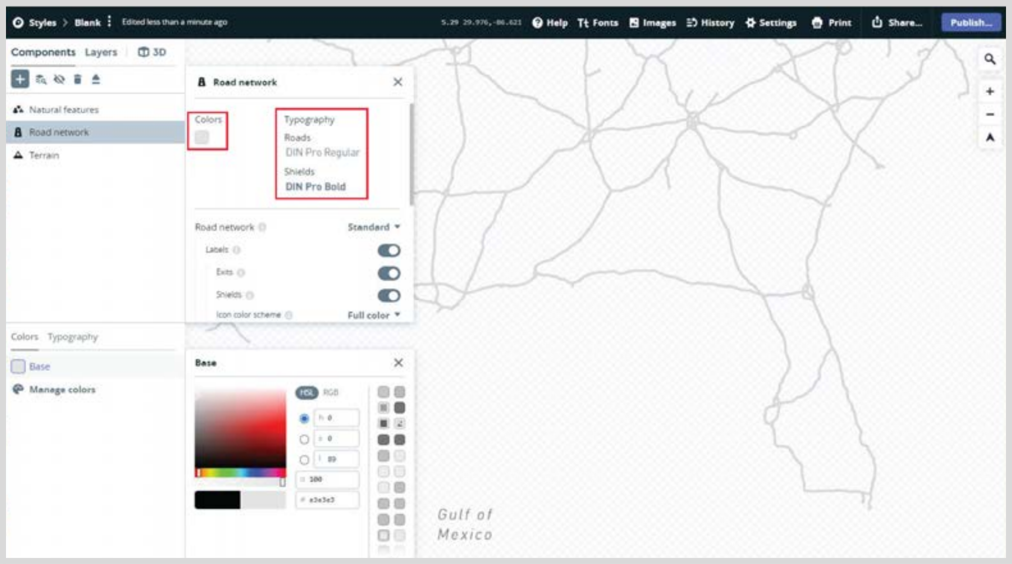

mapbox | Studio
Styles
Tilesets
Datasets
Choose a template
Basic
The best way to get started.
Monochrome
Start with a Monochrome style.
Streets
A complete basemap, perfect for incorporating your own data.
Outdoors
General basemap tailored to hiking, biking, and sport.
Satellite Streets
Global imagery enhanced with road and label hierarchy.
Navigation
Light and dark basemaps tailored to in-app navigation.
Blank ✓
Start a style from scratch.
Cancel
Customize Blank

2b Now, add the following **Components**:

- Land, water, & sky
- Natural features
- Place labels
- Road network
- Terrain



Play around with different styling options by clicking on these components and scrolling down to see the options. **Note: Label components are separated from Line and Polygon components**. This means that you will style the label (font, color, etc.) separately from the polygons that they refer to.



Road network

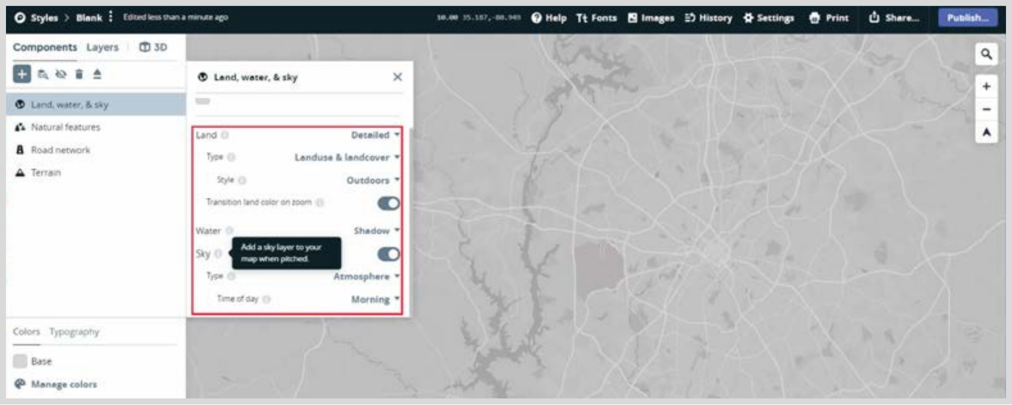
Colors

Typography
Roads: DIN Pro Regular
Shields: DIN Pro Bold

Road network: Standard
Labels: on
Exits: on
Shields: on
Icon color scheme: Full color

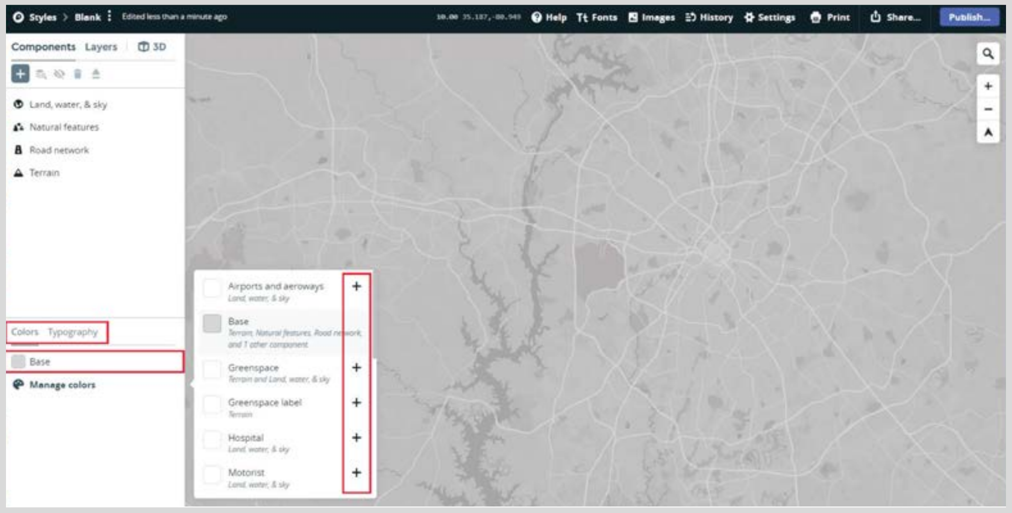
Base

HSL RGB
h 0
s 0
l 89
100
# e3e3e3



Land, water, & sky

Land: Detailed
Type: Landuse & landcover
Style: Outdoors
Transition land color on zoom: on
Water: Shadow
Sky: on
Add a sky layer to your map when pitched.
Type: Atmosphere
Time of day: Morning



Airports and aeroways
Land, water, & sky

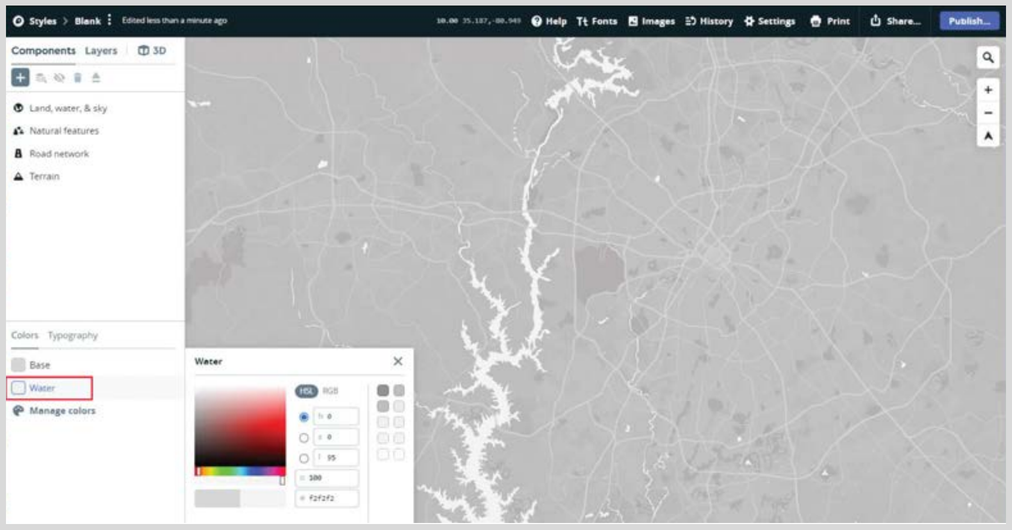
Base
Terrain, Natural features, Road network, and 1 other component

Greenspace
Terrain and Land, water, & sky

Greenspace label
Terrain

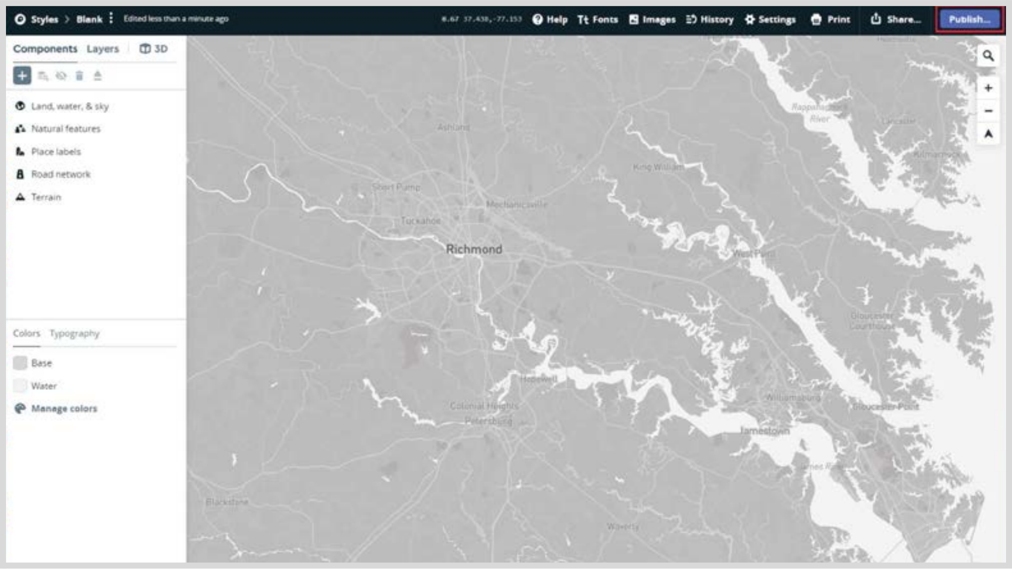
Hospital
Land, water, & sky

Motorist
Land, water, & sky



Mapbox's user interface has some redundancy baked in; that is, you can sometimes edit the same thing in two different places. As an example, you'll see **Typography** and **Color** tabs at the bottom of the Component editor area. Think of these as overall controls – you can control the base color of the entire map, or of certain general features (water, roads, etc.); likewise, you can change the fonts and colors of categories of labels. To change more granular information (eg. creeks or streams verses rivers in the "water" category), you will edit the specific component or its layers.

2c **Publish** your new Map style. This makes your changes live on the web, so you can see your updated map when you embed it. Note that when you make changes to your map, you will not be able to see these changes in your embedded map until you've Published them.

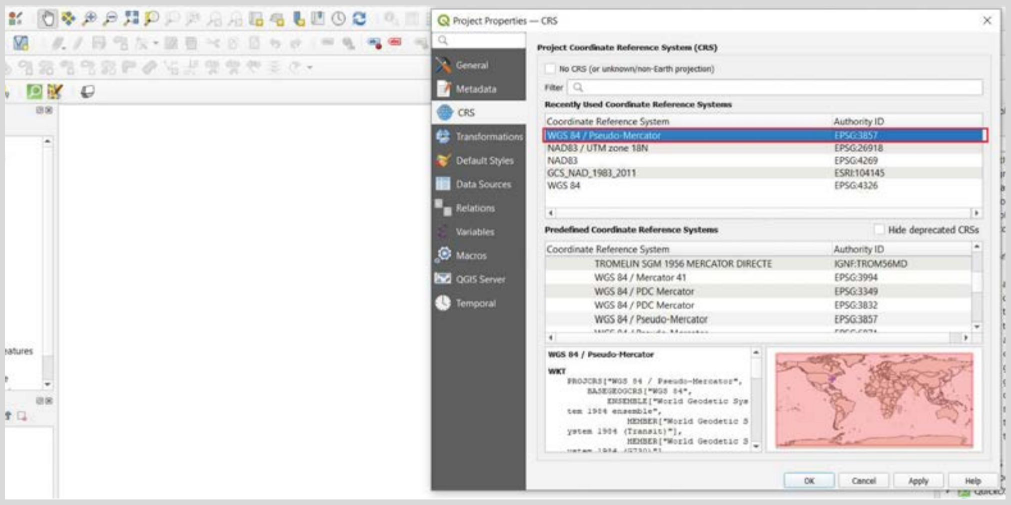


GIS for Designers
VT | A+D | SU22

Step 3: Export TIGER county data from QGIS for your Appalachian city's state.

Note: this is a larger scale than we've been dealing with (we've been using census block groups). Because we're uploading data for the entire state, it will be easier to see patterns at the county level.

3a Create a new QGIS file. Change the CRS to **EPSG 3857: Pseudo Mercador**. This is Mapbox's (and Google Maps's) projection system.



Project Coordinate Reference System (CRS)

□ No CRS (or unknown/non-Earth projection)

Filter

Recently Used Coordinate Reference Systems

Coordinate Reference System	Authority ID
WGS 84 / Pseudo-Mercator	EPSG:3857
NAD83 / UTM zone 18N	EPSG:26918
NAD83	EPSG:4269
GCS_NAD_1983_2011	ESRI:104145
WGS 84	EPSG:4326

Predefined Coordinate Reference Systems □ Hide deprecated CRSs

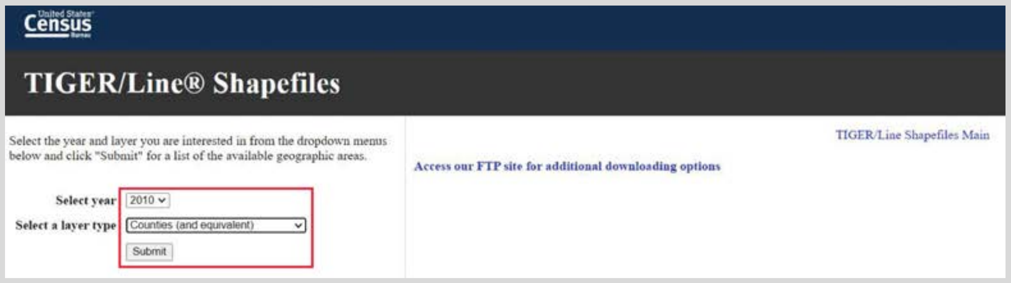
Coordinate Reference System	Authority ID
TROMELIN SGM 1956 MERCATOR DIRECTE	IGNF:TROM56MD
WGS 84 / Mercator 41	EPSG:3994
WGS 84 / PDC Mercator	EPSG:3349
WGS 84 / PDC Mercator	EPSG:3832
WGS 84 / Pseudo-Mercator	EPSG:3857

WGS 84 / Pseudo-Mercator

WKT

```
PROJCRS["WGS 84 / Pseudo-Mercator",
    BASEGEOGCRS["WGS 84",
        ENSEMBLE["World Geodetic Sys
tem 1984 ensemble",
            MEMBER["World Geodetic S
ystem 1984 (Transit)"],
            MEMBER["World Geodetic S
```

3b Download the **2010 TIGER shapefile for counties in your city's state:**
https://www.census.gov/geographies/mapping-files/time-series/geo/tiger-line-file.htm



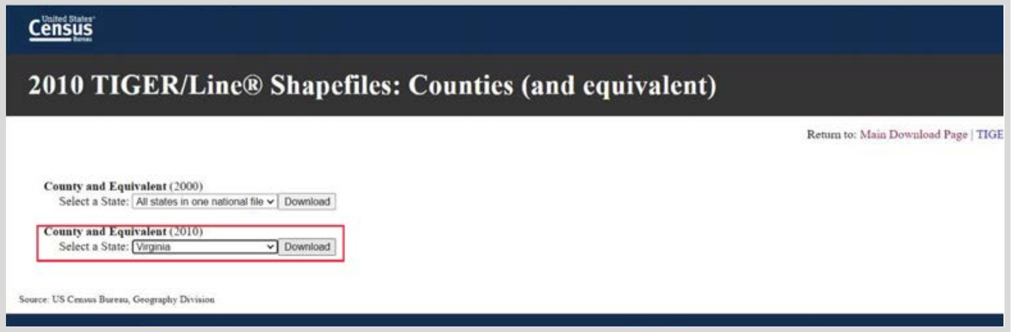
TIGER/Line® Shapefiles

Select the year and layer you are interested in from the dropdown menus below and click "Submit" for a list of the available geographic areas.

Access our FTP site for additional downloading options

Select year: 2010

Select a layer type: Counties (and equivalent)


United States Census Bureau
Return to: Main Download Page | TIGE
Download
Download

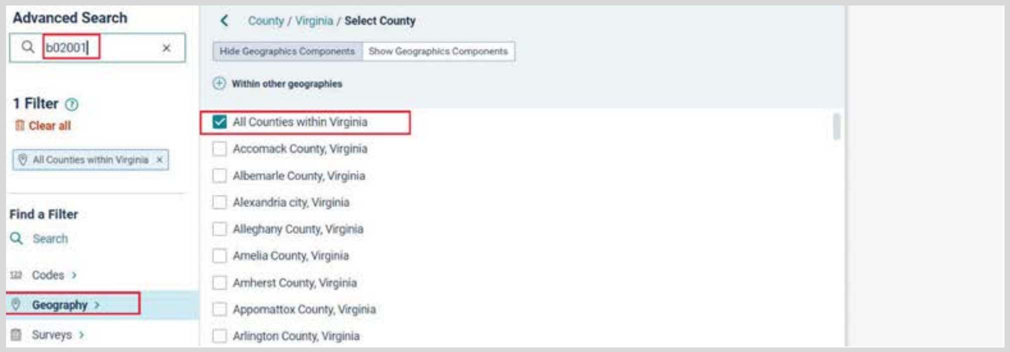
2010 TIGER/Line® Shapefiles: Counties (and equivalent)

County and Equivalent (2000)
Select a State: All states in one national file

County and Equivalent (2010)
Select a State: Virginia

Source: US Census Bureau, Geography Division

3c Download BOTH the **2013 and 2019** county ACS tables **B02001** (racial): https://data.census.gov/cedsci/



Advanced Search

b02001

1 Filter
Clear all
All Counties within Virginia ×

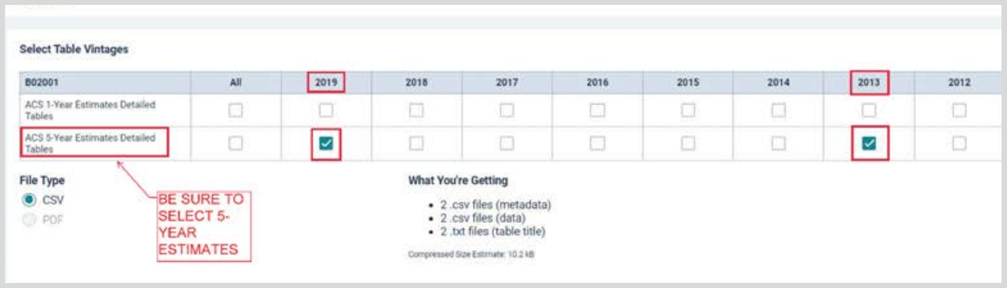
Find a Filter
Search
Codes ›
Geography ›
Surveys ›

County / Virginia / **Select County**

Hide Geographics Components | Show Geographics Components

Within other geographies

- ☑ All Counties within Virginia
- ☐ Accomack County, Virginia
- ☐ Albemarle County, Virginia
- ☐ Alexandria city, Virginia
- ☐ Alleghany County, Virginia
- ☐ Amelia County, Virginia
- ☐ Amherst County, Virginia
- ☐ Appomattox County, Virginia
- ☐ Arlington County, Virginia



Select Table Vintages

B02001	All	2019	2018	2017	2016	2015	2014	2013	2012
ACS 1-Year Estimates Detailed Tables	☐	☐	☐	☐	☐	☐	☐	☐	☐
ACS 5-Year Estimates Detailed Tables	☐	☑	☐	☐	☐	☐	☐	☑	☐

File Type
- ● CSV
- ○ PDF

BE SURE TO SELECT 5-YEAR ESTIMATES

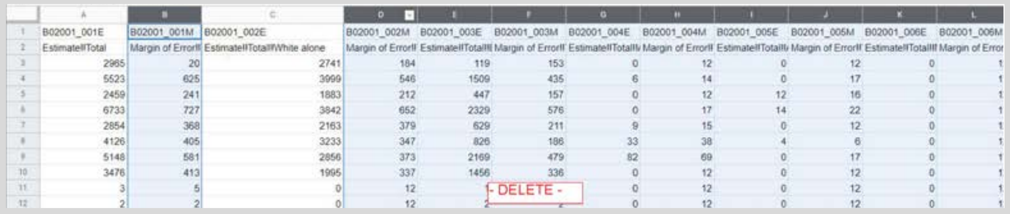
What You're Getting
- 2 .csv files (metadata)
- 2 .csv files (data)
- 2 .txt files (table title)

Compressed Size Estimate: 10.2 kB

NOTE: **UNZIP** your downloaded folders before doing anything else with them. Do not try to open or import the shapefile or census tables without unzipping them first. Be sure to save your unzipped files in your class folder.

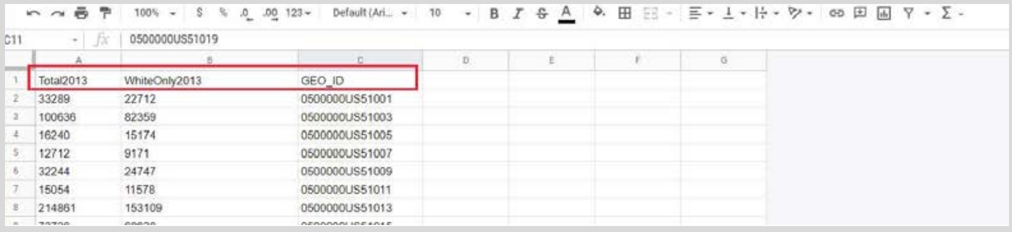
3d Clean up your ACS data and find the correct GEOIDs to match the TIGER file. To simplify this, check out the GEOID attribute table of your shapefile verses your ACS GEOIDs, and you'll notice that the leading string is always the same: **0500000US**. You can simply **Find and Replace** this with nothing in Google Sheets to remove it from all your GEOIDs (which will then match the TIGER file).

NOTE: be sure to include the year in your ACS column names (eg **WhiteOnly2019** or **WhiteOnly2013**).



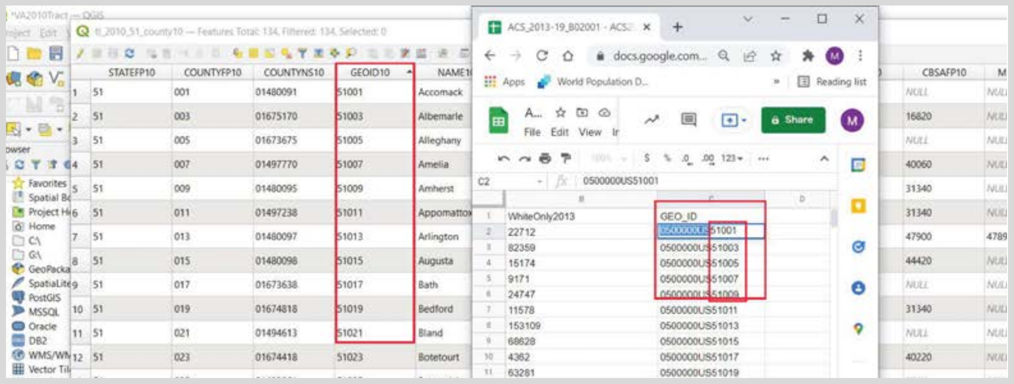
	A	B	C	D	E	F	G	H	I	J	K	L
1	B02001_001E	B02001_001M	B02001_002E	B02001_002M	B02001_003E	B02001_003M	B02001_004E	B02001_004M	B02001_005E	B02001_005M	B02001_006E	B02001_006M
2	Estimate!!Total	Margin of Error!!	Estimate!!Total!!White alone	Margin of Error!!	Estimate!!Total!!!	Margin of Error!!	Estimate!!Total!!\	Margin of Error!!	Estimate!!Total!!\	Margin of Error!!	Estimate!!Total!!!	Margin of Error
3	2965	20	2741	184	119	153	0	12	0	12	0	1
4	5523	625	3999	546	1509	435	6	14	0	17	0	1
5	2459	241	1883	212	447	157	0	12	12	16	0	1
6	6733	727	3842	652	2329	576	0	17	14	22	0	1
7	2854	368	2163	379	629	211	9	15	0	12	0	1
8	4126	405	3233	347	826	186	33	38	4	6	0	1
9	5148	581	2856	373	2169	479	82	69	0	17	0	1
10	3476	413	1995	337	1456	336	0	12	0	12	0	1
11	3	5	0	12	1		0	12	0	12	0	1
12	2	2	0	12	2		0	12	0	12	0	1

- DELETE -



C11 | fx | 0500000US51019

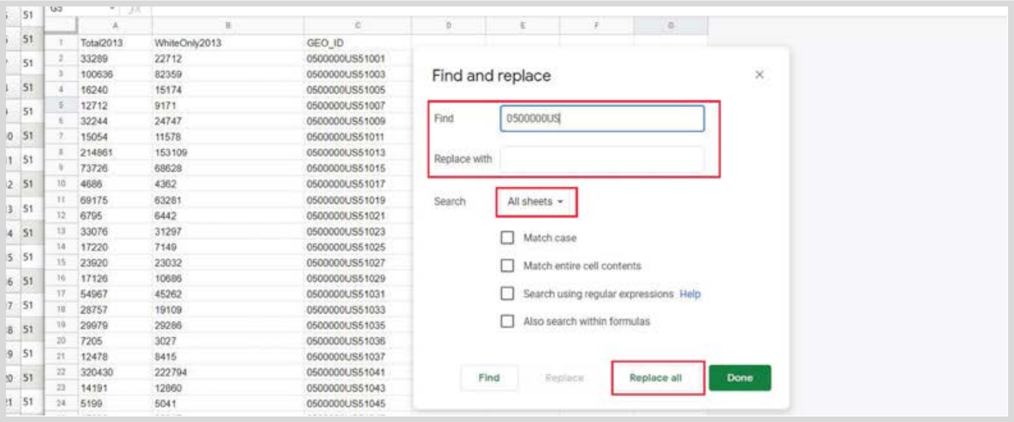
	A	B	C	D	E	F	G
1	Total2013	WhiteOnly2013	GEO_ID				
2	33289	22712	0500000US51001				
3	100636	82359	0500000US51003				
4	16240	15174	0500000US51005				
5	12712	9171	0500000US51007				
6	32244	24747	0500000US51009				
7	15054	11578	0500000US51011				
8	214861	153109	0500000US51013				


*VA2010Tract — QGIS
tl_2010_51_county10 — Features Total: 134, Filtered: 134, Selected: 0
ACS_2013-19_B02001 - ACS
docs.google.com...
Share

	STATEFP10	COUNTYFP10	COUNTYNS10	GEOID10	NAME10	CBSAFP10
1	51	001	01480091	51001	Accomack	NULL
2	51	003	01675170	51003	Albemarle	16820
3	51	005	01673675	51005	Alleghany	NULL
4	51	007	01497770	51007	Amelia	40060
5	51	009	01480095	51009	Amherst	31340
6	51	011	01497238	51011	Appomattox	31340
7	51	013	01480097	51013	Arlington	47900
8	51	015	01480098	51015	Augusta	44420
9	51	017	01673638	51017	Bath	NULL
10	51	019	01674818	51019	Bedford	31340
11	51	021	01494613	51021	Bland	NULL
12	51	023	01674418	51023	Botetourt	40220

C2 0500000US51001

	B	C
1	WhiteOnly2013	GEO_ID
2	22712	0500000US51001
3	82359	0500000US51003
4	15174	0500000US51005
5	9171	0500000US51007
6	24747	0500000US51009
7	11578	0500000US51011
8	153109	0500000US51013
9	68628	0500000US51015
10	4362	0500000US51017
11	63281	0500000US51019



	A	B	C
1	Total2013	WhiteOnly2013	GEO_ID
2	33289	22712	0500000US51001
3	100636	82359	0500000US51003
4	16240	15174	0500000US51005
5	12712	9171	0500000US51007
6	32244	24747	0500000US51009
7	15054	11578	0500000US51011
8	214861	153109	0500000US51013
9	73726	68628	0500000US51015
10	4686	4362	0500000US51017
11	69175	63281	0500000US51019
12	6795	6442	0500000US51021
13	33076	31297	0500000US51023
14	17220	7149	0500000US51025
15	23920	23032	0500000US51027
16	17126	10686	0500000US51029
17	54967	45262	0500000US51031
18	28757	19109	0500000US51033
19	29979	29286	0500000US51035
20	7205	3027	0500000US51036
21	12478	8415	0500000US51037
22	320430	222794	0500000US51041
23	14191	12860	0500000US51043
24	5199	5041	0500000US51045

Find and replace

Find: 0500000US

Replace with:

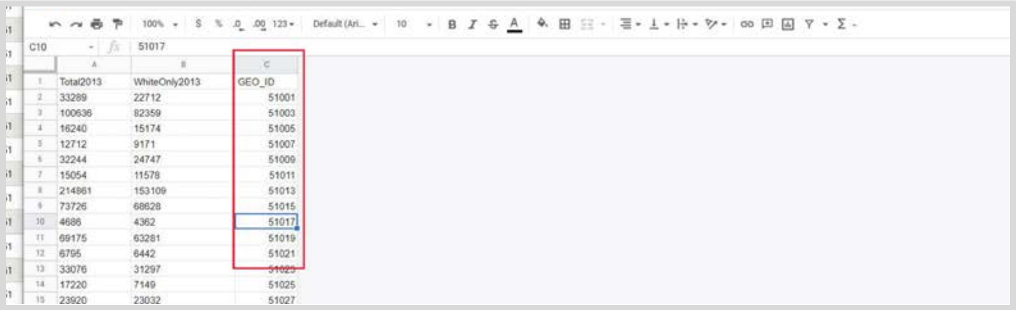
Search: All sheets

□ Match case

□ Match entire cell contents

□ Search using regular expressions Help

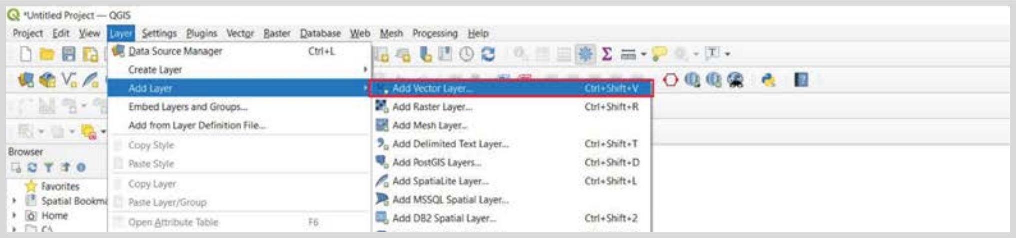
□ Also search within formulas

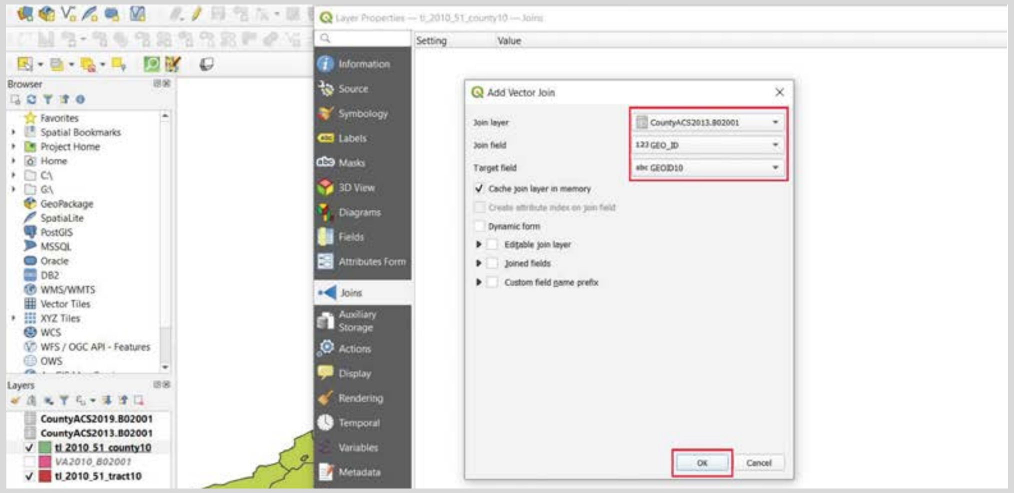


C10 51017

	A	B	C
1	Total2013	WhiteOnly2013	GEO_ID
2	33289	22712	51001
3	100636	82359	51003
4	16240	15174	51005
5	12712	9171	51007
6	32244	24747	51009
7	15054	11578	51011
8	214861	153109	51013
9	73726	68628	51015
10	4686	4362	51017
11	69175	63281	51019
12	6795	6442	51021
13	33076	31297	51023
14	17220	7149	51025
15	23920	23032	51027

3e **Import** the **TIGER file and ACS .csvs** to QGIS, and join **BOTH** the 2013 and 2019 tables to your TIGER shapefile, and check that the join was successful in the Attribute Table.


*Untitled Project — QGIS
Project Edit View Layer Settings Plugins Vector Raster Database Web Mesh Processing Help
Data Source Manager Ctrl+L
Create Layer
Add Layer
Embed Layers and Groups...
Add from Layer Definition File...
Copy Style
Paste Style
Copy Layer
Paste Layer/Group
Open Attribute Table F6
Add Vector Layer... Ctrl+Shift+V
Add Raster Layer... Ctrl+Shift+R
Add Mesh Layer...
Add Delimited Text Layer... Ctrl+Shift+T
Add PostGIS Layers... Ctrl+Shift+D
Add SpatiaLite Layer... Ctrl+Shift+L
Add MSSQL Spatial Layer...
Add DB2 Spatial Layer... Ctrl+Shift+2
Browser
Favorites
Spatial Bookmarks
Home



Layer Properties — tl_2010_51_county10 — Joins

Add Vector Join

Join layer: CountyACS2013.B02001

Join field: 123 GEO_ID

Target field: abc GEOID10

☑ Cache join layer in memory

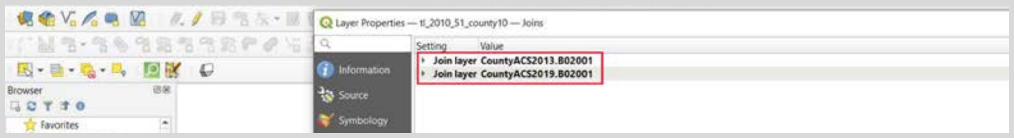
□ Create attribute index on join field

□ Dynamic form

□ Editable join layer

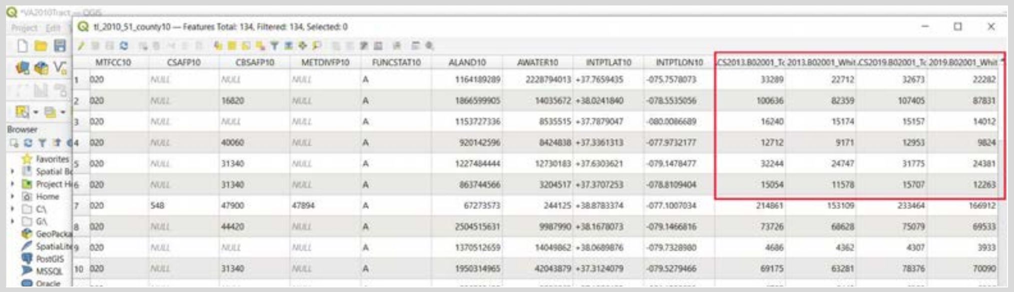
□ Joined fields

□ Custom field name prefix



Layer Properties — tl_2010_51_county10 — Joins

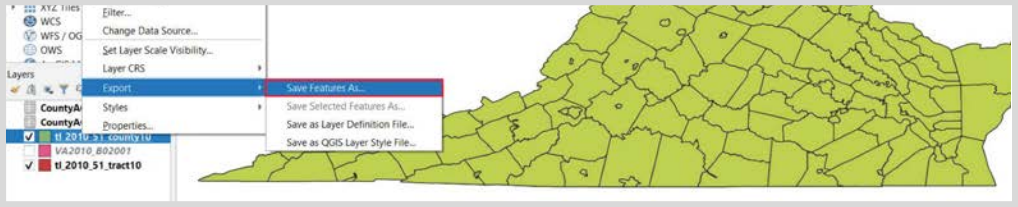
Setting	Value
Join layer CountyACS2013.B02001	
Join layer CountyACS2019.B02001	

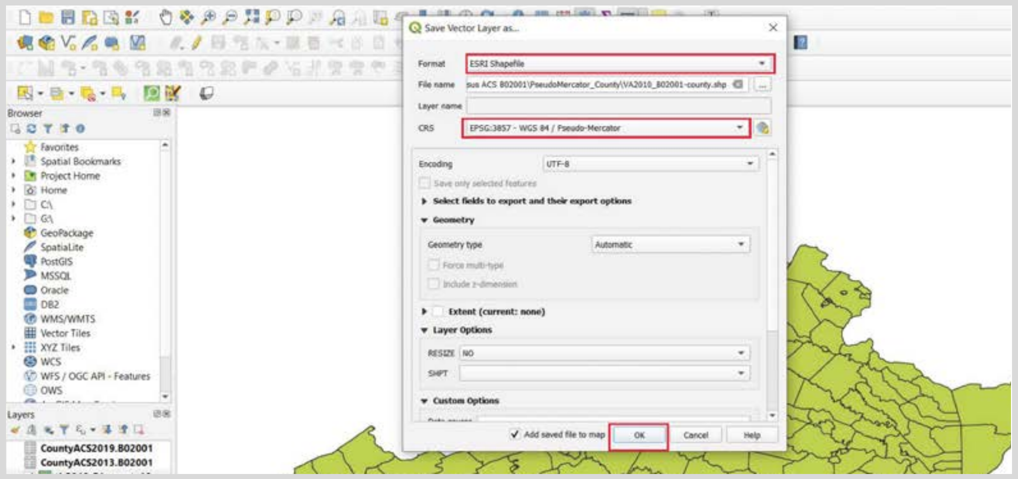


tl_2010_51_county10 — Features Total: 134, Filtered: 134, Selected: 0

	MTFCC10	CSAFP10	CBSAFP10	METDIVFP10	FUNCSTAT10	ALAND10	AWATER10	INTPTLAT10	INTPTLON10	CS2013.B02001_Tc	2013.B02001_Whit	CS2019.B02001_Tc	2019.B02001_Whit
1	020	NULL	NULL	NULL	A	1164189289	2228794013	+37.7659435	-075.7578073	33289	22712	32673	22282
2	020	NULL	16820	NULL	A	1866599905	14035672	+38.0241840	-078.5535056	100636	82359	107405	87831
3	020	NULL	NULL	NULL	A	1153727336	8535515	+37.7879047	-080.0086689	16240	15174	15157	14012
4	020	NULL	40060	NULL	A	920142596	8424838	+37.3361313	-077.9732177	12712	9171	12953	9824
5	020	NULL	31340	NULL	A	1227484444	12730183	+37.6303621	-079.1478477	32244	24747	31775	24381
6	020	NULL	31340	NULL	A	863744566	3204517	+37.3707253	-078.8109404	15054	11578	15707	12263
7	020	548	47900	47894	A	67273573	244125	+38.8783374	-077.1007034	214861	153109	233464	166912
8	020	NULL	44420	NULL	A	2504515631	9987990	+38.1678073	-079.1466816	73726	68628	75079	69533
9	020	NULL	NULL	NULL	A	1370512659	14049862	+38.0689876	-079.7328980	4686	4362	4307	3933
10	020	NULL	31340	NULL	A	1950314965	42043879	+37.3124079	-079.5279466	69175	63281	78376	70090

3f Export the joined shapefile as an “ESRI Shapefile”, saved in a new folder so you can easily zip its various parts. Make sure to set your **export** CRS to “**EPSG 3857 PseudoMercator**”. This is Mapbox’s projection system





Save Vector Layer as...

Format: ESRI Shapefile

File name: sus ACS B02001\PseudoMercator_County\VA2010_B02001-county.shp

Layer name:

CRS: EPSG:3857 - WGS 84 / Pseudo-Mercator

Encoding: UTF-8

☐ Save only selected features

Geometry type: Automatic

☐ Force multi-type

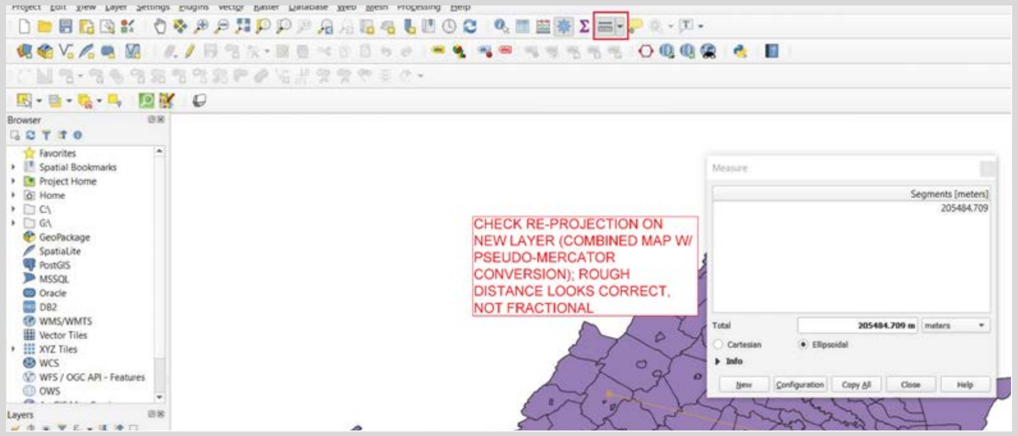
☐ Include z-dimension

☐ Extent (current: none)

RESIZE: NO

SHPT:

☑ Add saved file to map



Measure

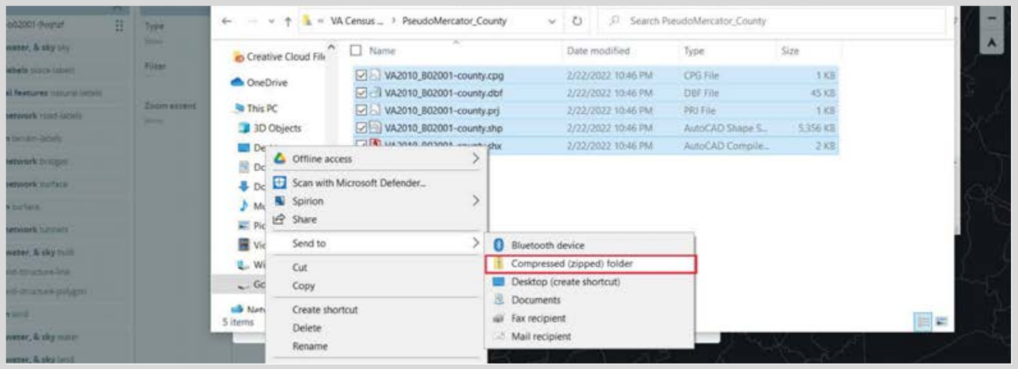
Segments [meters]: 205484.709

Total: 205484.709 m meters

○ Cartesian ● Ellipsoidal

CHECK RE-PROJECTION ON NEW LAYER (COMBINED MAP W/ PSEUDO-MERCATOR CONVERSION); ROUGH DISTANCE LOOKS CORRECT, NOT FRACTIONAL

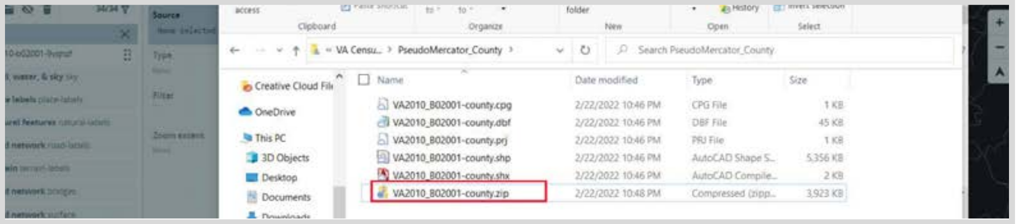
3g **Zip** your Shapefile files.



VA Census ... › PseudoMercator_County

Name	Date modified	Type	Size
☑ VA2010_B02001-county.cpg	2/22/2022 10:46 PM	CPG File	1 KB
☑ VA2010_B02001-county.dbf	2/22/2022 10:46 PM	DBF File	45 KB
☑ VA2010_B02001-county.prj	2/22/2022 10:46 PM	PRJ File	1 KB
☑ VA2010_B02001-county.shp	2/22/2022 10:46 PM	AutoCAD Shape S...	5,356 KB
VA2010_B02001-county.shx	2/22/2022 10:46 PM	AutoCAD Compile...	2 KB

5 items

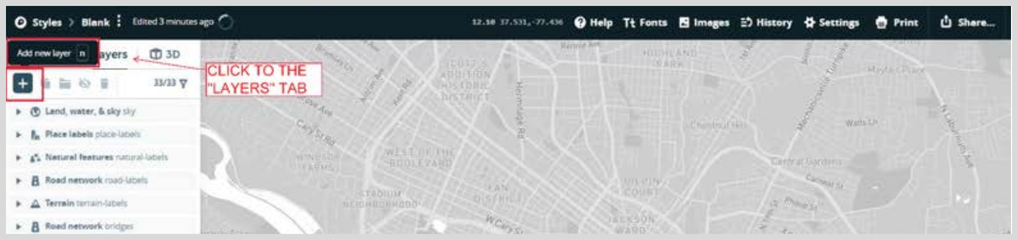


VA Censu... › PseudoMercator_County ›

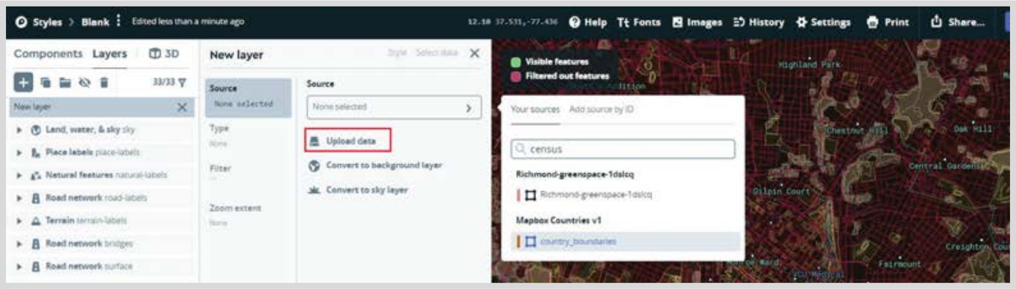
Name	Date modified	Type	Size
VA2010_B02001-county.cpg	2/22/2022 10:46 PM	CPG File	1 KB
VA2010_B02001-county.dbf	2/22/2022 10:46 PM	DBF File	45 KB
VA2010_B02001-county.prj	2/22/2022 10:46 PM	PRJ File	1 KB
VA2010_B02001-county.shp	2/22/2022 10:46 PM	AutoCAD Shape S...	5,356 KB
VA2010_B02001-county.shx	2/22/2022 10:46 PM	AutoCAD Compile...	2 KB
VA2010_B02001-county.zip	2/22/2022 10:48 PM	Compressed (zipp...	3,923 KB

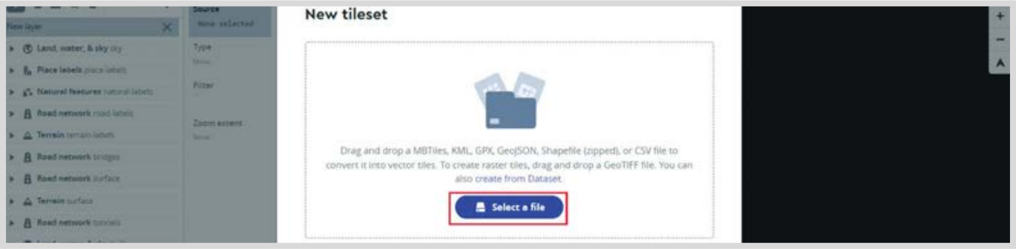
Step 4: Add your zipped Shapefile as a layer to your Mapbox base map.

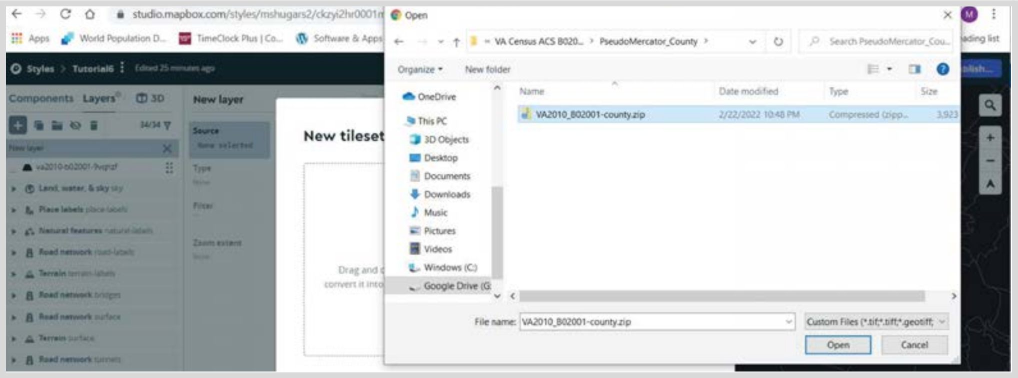
4a Under the **Layers tab** in your Studio view, click the **plus** sign “**Add New Layer**”

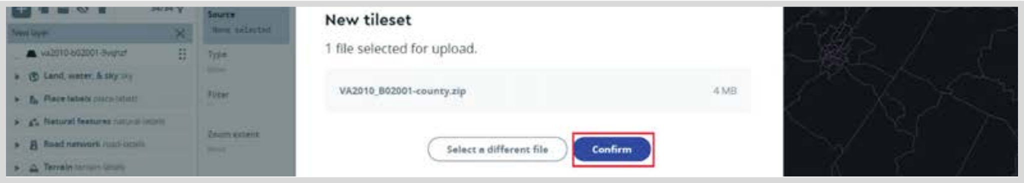


4b In Source, select “**Upload Data**” and select your **zipped shapefile**.

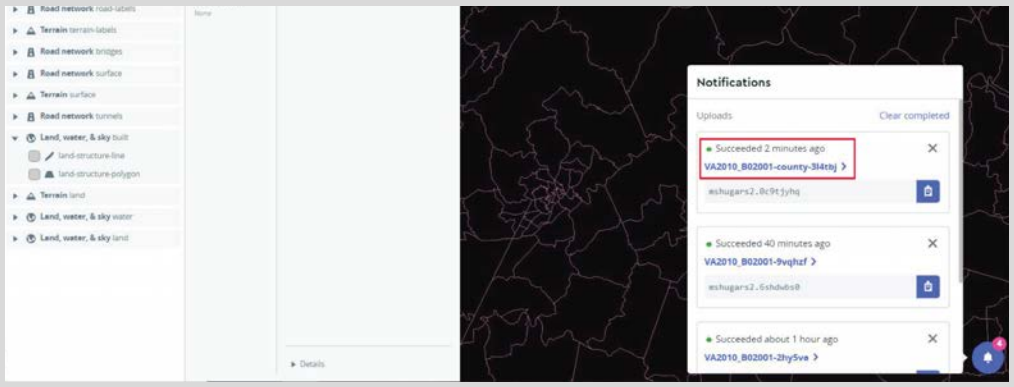



New tileset
Drag and drop a MBTiles, KML, GPX, GeoJSON, Shapefile (zipped), or CSV file to convert it into vector tiles. To create raster tiles, drag and drop a GeoTIFF file. You can also create from Dataset.
Select a file





4c Once the upload has processed (check the lower right of your screen), scroll down to your newly uploaded tileset (**check the notification list for its name**), **click** on it, and select the **square shapefile symbol sublayer.** You should see the outline of your state show up in bright green.



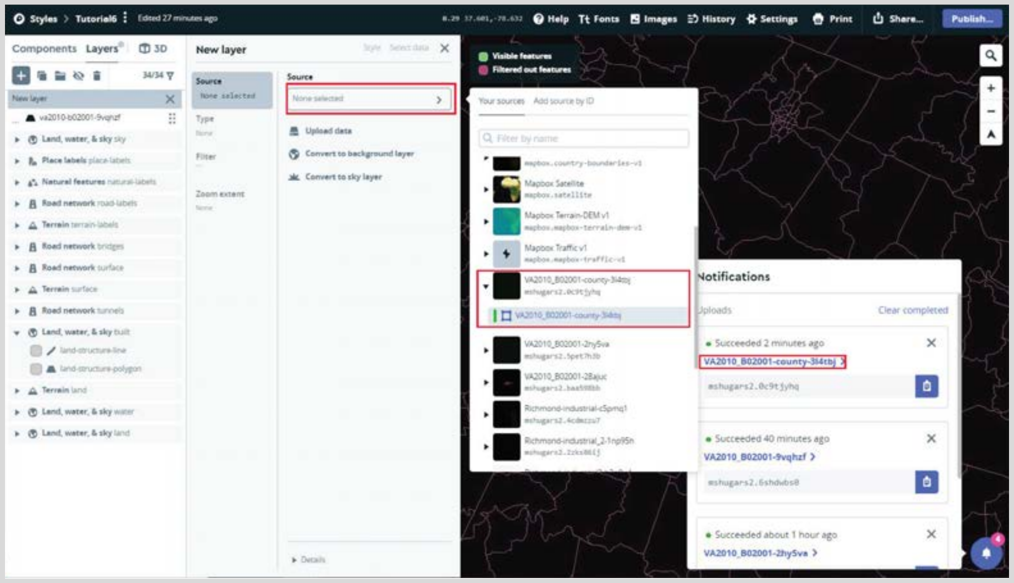
Notifications

Uploads — Clear completed

● Succeeded 2 minutes ago
VA2010_B02001-county-3l4tbj ›
mshugars2.0c9tjyhq

● Succeeded 40 minutes ago
VA2010_B02001-9vqhzf ›
mshugars2.6shdwbs0

● Succeeded about 1 hour ago
VA2010_B02001-2hy5va ›



mapbox.country-boundaries-v1

Mapbox Satellite
mapbox.satellite

Mapbox Terrain-DEM v1
mapbox.mapbox-terrain-dem-v1

Mapbox Traffic v1
mapbox.mapbox-traffic-v1

VA2010_B02001-county-3l4tbj
mshugars2.0c9tjyhq
VA2010_B02001-county-3l4tbj

VA2010_B02001-2hy5va
mshugars2.5pet7h3b

VA2010_B02001-28ajuc
mshugars2.baa59Bbb

Richmond-industrial-c5pmq1
mshugars2.4cdmzzu7

Richmond-industrial_2-1np95n
mshugars2.2zks86ij

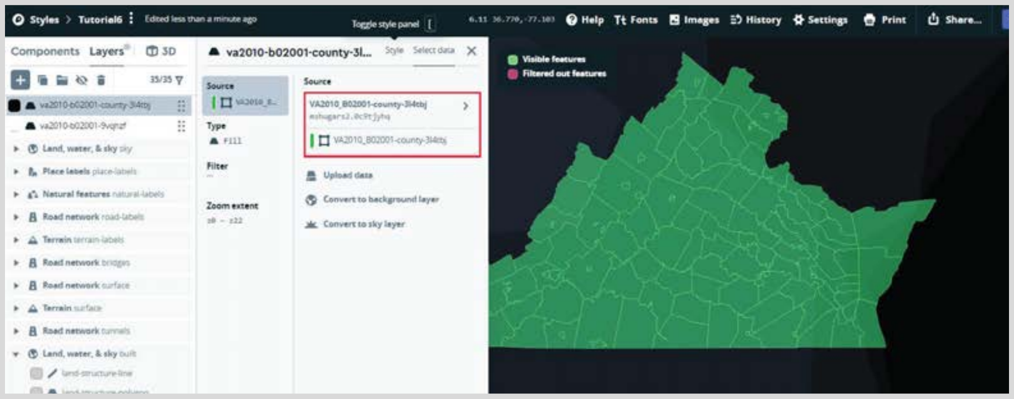
Notifications

Uploads — Clear completed

● Succeeded 2 minutes ago
VA2010_B02001-county-3l4tbj ›
mshugars2.0c9tjyhq

● Succeeded 40 minutes ago
VA2010_B02001-9vqhzf ›
mshugars2.6shdwbs0

● Succeeded about 1 hour ago
VA2010_B02001-2hy5va ›



Source: VA2010_B...

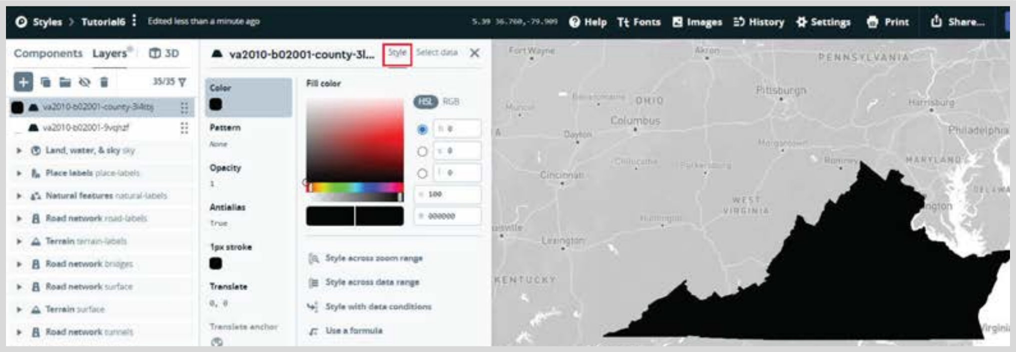
Type: Fill

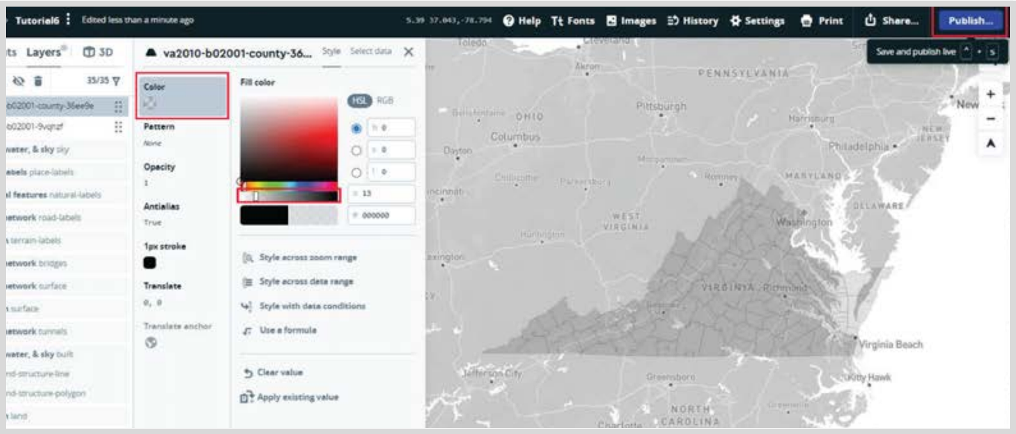
Filter: —

Zoom extent: z0 – z22

Source
VA2010_B02001-county-3l4tbj
mshugars2.0c9tjyhq
VA2010_B02001-county-3l4tbj

4d Click the “**Style**” tab, and for now simply slide the fill color opacity to around 20%. In the next Tutorial, we’ll do fancier styling on this layer. For now, you want to see the ghostly shape of the state with its counties outlined in black.



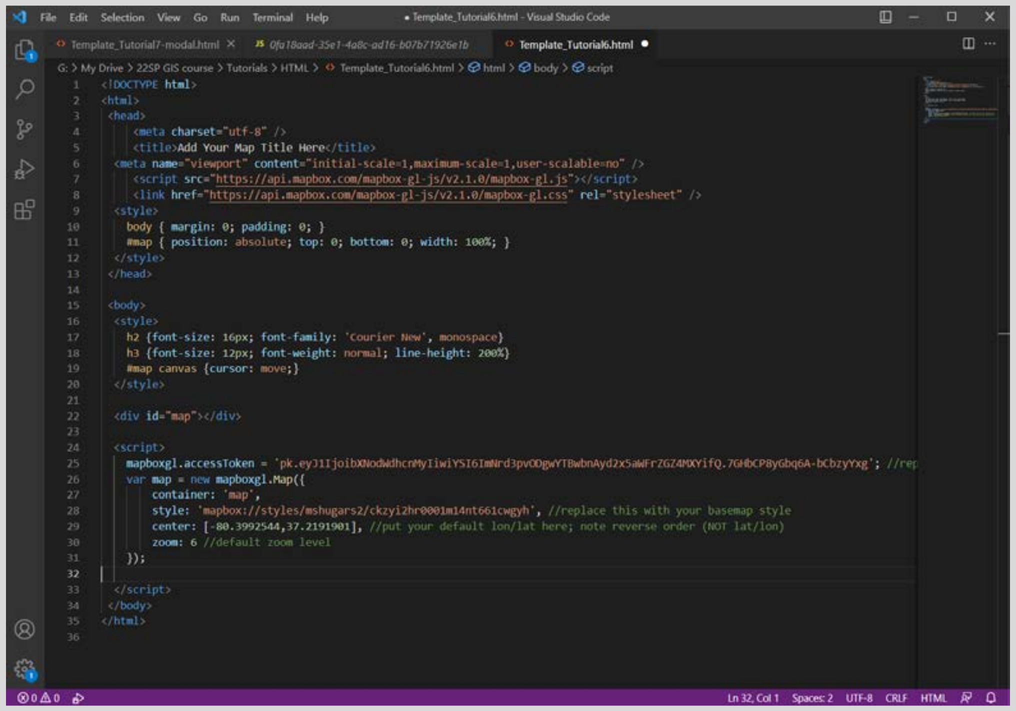


4e **Publish** your basemap changes (upper right button).

Step 5: Download and install Visual Studio Code (this will be our code editor):

https://code.visualstudio.com/download

5a Download the **HTML template** (Canvas > Files > Templates > “Template-Tutorial6-7”) for this tutorial and open it in Visual Studio:

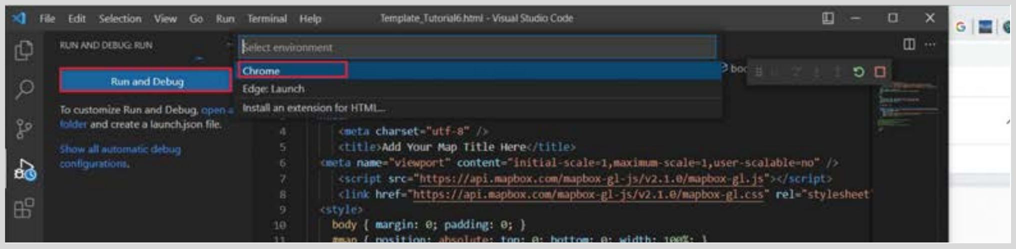


File Edit Selection View Go Run Terminal Help
• Template_Tutorial6.html - Visual Studio Code
Template_Tutorial7-modal.html
0fa18aad-35e1-4a8c-ad16-b07b71926e1b
Template_Tutorial6.html
G: > My Drive > 22SP GIS course > Tutorials > HTML > Template_Tutorial6.html > html > body > script
Ln 32, Col 1 Spaces: 2 UTF-8 CRLF HTML

```
<!DOCTYPE html>
<html>
 <head>
     <meta charset="utf-8" />
     <title>Add Your Map Title Here</title>
  <meta name="viewport" content="initial-scale=1,maximum-scale=1,user-scalable=no" />
     <script src="https://api.mapbox.com/mapbox-gl-js/v2.1.0/mapbox-gl.js"></script>
     <link href="https://api.mapbox.com/mapbox-gl-js/v2.1.0/mapbox-gl.css" rel="stylesheet" />
  <style>
    body { margin: 0; padding: 0; }
    #map { position: absolute; top: 0; bottom: 0; width: 100%; }
  </style>
 </head>

 <body>
  <style>
    h2 {font-size: 16px; font-family: 'Courier New', monospace}
    h3 {font-size: 12px; font-weight: normal; line-height: 200%}
    #map canvas {cursor: move;}
  </style>

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

  <script>
    mapboxgl.accessToken = 'pk.eyJ1IjoibXNodWdhcnMyIiwiYSI6ImNrd3pvODgwYTBwbnAyd2x5aWFrZGZ4MXYifQ.7GHbCP8yGbq6A-bCbzyYxg'; //rep
    var map = new mapboxgl.Map({
        container: 'map',
        style: 'mapbox://styles/mshugars2/ckzyi2hr0001m14nt661cwgyh', //replace this with your basemap style
        center: [-80.3992544,37.2191901], //put your default lon/lat here; note reverse order (NOT lat/lon)
        zoom: 6 //default zoom level
    });

  </script>
 </body>
</html>
```

Step 6: Update HTML code with your map.

6a First, run your HTML in Chrome to see what it looks like by default. Reload this page frequently to check the changes you've made:

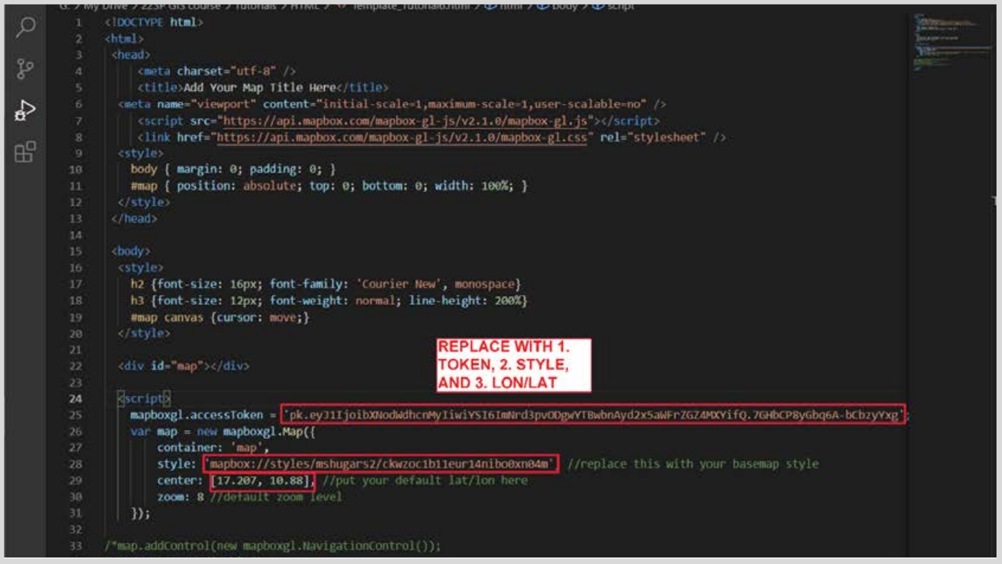


To customize Run and Debug, open a folder and create a launch.json file.

Show all automatic debug configurations.

```
        <meta charset="utf-8" />
        <title>Add Your Map Title Here</title>
    <meta name="viewport" content="initial-scale=1,maximum-scale=1,user-scalable=no" />
        <script src="https://api.mapbox.com/mapbox-gl-js/v2.1.0/mapbox-gl.js"></script>
        <link href="https://api.mapbox.com/mapbox-gl-js/v2.1.0/mapbox-gl.css" rel="stylesheet"
    <style>
      body { margin: 0; padding: 0; }
```

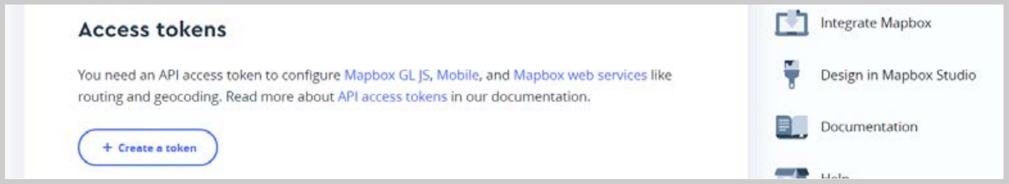
Next, you're going to update the **ACCESS TOKEN**, **MAP STYLE URL**, **CENTER, and ZOOM** in the HTML. Scroll down to this section (below the <*script*> tag):



```
<!DOCTYPE html>
<html>
 <head>
        <meta charset="utf-8" />
        <title>Add Your Map Title Here</title>
    <meta name="viewport" content="initial-scale=1,maximum-scale=1,user-scalable=no" />
        <script src="https://api.mapbox.com/mapbox-gl-js/v2.1.0/mapbox-gl.js"></script>
        <link href="https://api.mapbox.com/mapbox-gl-js/v2.1.0/mapbox-gl.css" rel="stylesheet" />
    <style>
      body { margin: 0; padding: 0; }
      #map { position: absolute; top: 0; bottom: 0; width: 100%; }
    </style>
 </head>

 <body>
    <style>
      h2 {font-size: 16px; font-family: 'Courier New', monospace}
      h3 {font-size: 12px; font-weight: normal; line-height: 200%}
      #map canvas {cursor: move;}
    </style>

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

    <script>
      mapboxgl.accessToken = 'pk.eyJ1IjoibXNodWdhcnMyIiwiYSI6ImNrd3pvODgwYTBwbnAyd2x5aWFrZGZ4MXYifQ.7GHbCP8yGbq6A-bCbzyYxg';
      var map = new mapboxgl.Map({
          container: 'map',
          style: 'mapbox://styles/mshugars2/ckwzoc1b11eur14nibo0xn04m' //replace this with your basemap style
          center: [17.207, 10.88], //put your default lat/lon here
          zoom: 8 //default zoom level
      });

  /*map.addControl(new mapboxgl.NavigationControl());
```

REPLACE WITH 1. TOKEN, 2. STYLE, AND 3. LON/LAT

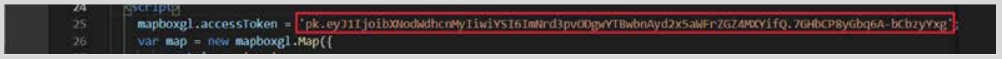
6b First, you need to create a Mapbox Access token from your Mapbox home page (https://account.mapbox.com/). Click "**Create a token**". Name it anything and confirm.



Access tokens

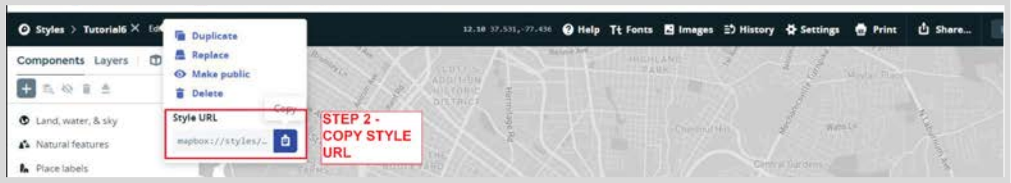
You need an API access token to configure Mapbox GL JS, Mobile, and Mapbox web services like routing and geocoding. Read more about API access tokens in our documentation.

Now, **copy** the new Token number (also known as an API keyl) to the HTML template, replacing the 'pk…' string **INSIDE the quotation marks** after *mapboxgl.accessToken*:



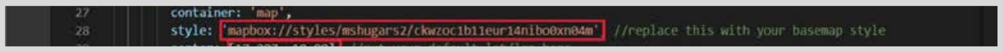
```
    <script>
      mapboxgl.accessToken = 'pk.eyJ1IjoibXNodWdhcnMyIiwiYSI6ImNrd3pvODgwYTBwbnAyd2x5aWFrZGZ4MXYifQ.7GHbCP8yGbq6A-bCbzyYxg';
      var map = new mapboxgl.Map({
```

6c Second, add your **Map Style URL** to the HTML template. You can find the map style (eg basemap) url back in your Mapbox Studio. There are several places to find the url, but one is in the Style editor as shown below:



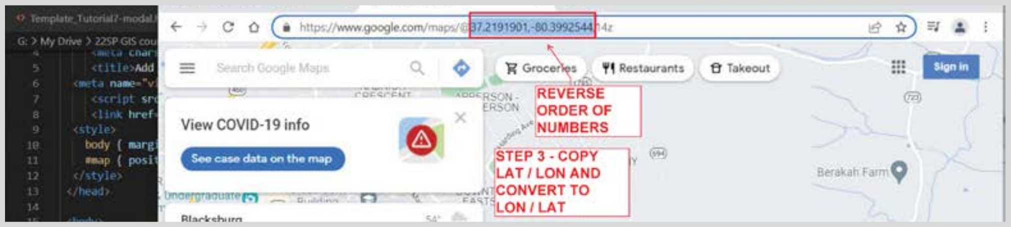
STEP 2 - COPY STYLE URL

Replace the text **INSIDE the quotation marks** after *style* in the template:



```
container: 'map',
style: 'mapbox://styles/mshugars2/ckwzoc1b11eur14nibo0xn04m' //replace this with your basemap style
```

6d Third, update the **Zoom level and Cente**r (Longitude / latitude starting point) in the template. Note that the template reads the center as Longitude / latitude, and NOT Latitude / longitude as it's shown in Google Maps. You need to **reverse** the numbers from Google Maps.

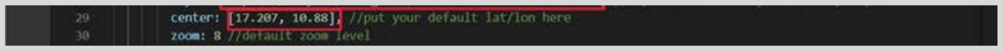


```
https://www.google.com/maps/@37.2191901,-80.3992544,14z
```

REVERSE ORDER OF NUMBERS

STEP 3 - COPY LAT / LON AND CONVERT TO LON / LAT

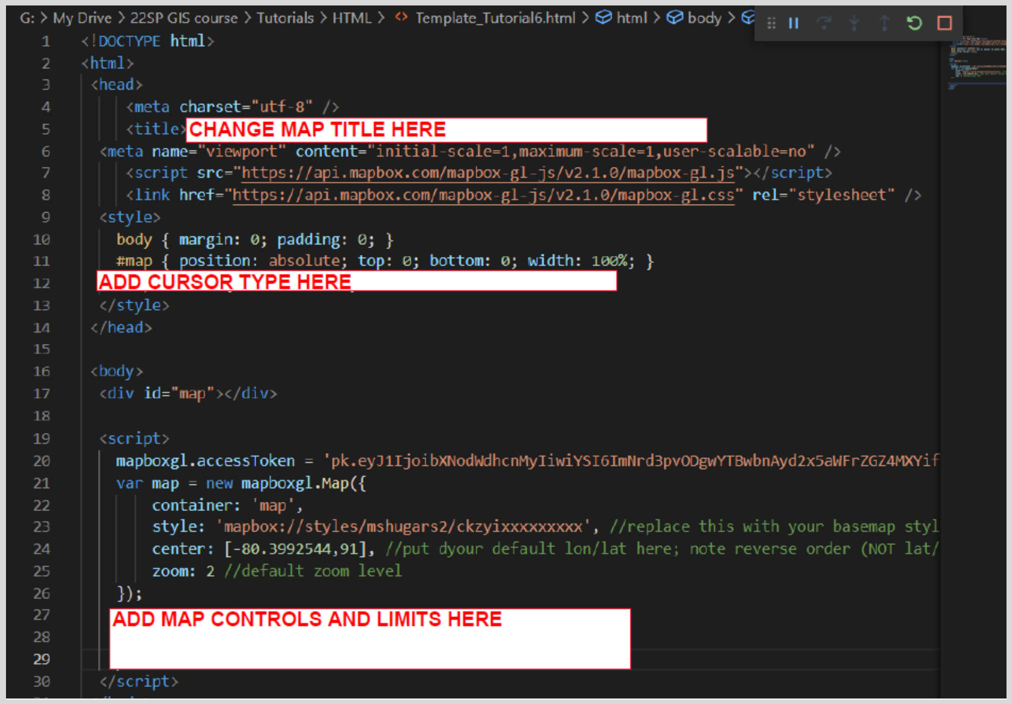
Replace the numbers after *center* in the template. Replace the number after *zoom* with an appropriate zoom level (try a few out to see what looks right by saving and reloading the map in Chrome).



```
center: [17.207, 10.88], //put your default lat/lon here
zoom: 8 //default zoom level
```

6e **Check** that the code still works and looks right by **saving the file and reloading the map in Chrome**.

Step 7: Some final map edits



G: > My Drive > 22SP GIS course > Tutorials > HTML > Template_Tutorial6.html > html > body >

```
<!DOCTYPE html>
<html>
 <head>
    <meta charset="utf-8" />
    <title>CHANGE MAP TITLE HERE
  <meta name="viewport" content="initial-scale=1,maximum-scale=1,user-scalable=no" />
    <script src="https://api.mapbox.com/mapbox-gl-js/v2.1.0/mapbox-gl.js"></script>
    <link href="https://api.mapbox.com/mapbox-gl-js/v2.1.0/mapbox-gl.css" rel="stylesheet" />
  <style>
    body { margin: 0; padding: 0; }
    #map { position: absolute; top: 0; bottom: 0; width: 100%; }
  ADD CURSOR TYPE HERE
  </style>
 </head>

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

  <script>
    mapboxgl.accessToken = 'pk.eyJ1IjoibXNodWdhcnMyIiwiYSI6ImNrd3pvODgwYTBwbnAyd2x5aWFrZGZ4MXYif
    var map = new mapboxgl.Map({
        container: 'map',
        style: 'mapbox://styles/mshugars2/ckzyixxxxxxxxx', //replace this with your basemap styl
        center: [-80.3992544,91], //put dyour default lon/lat here; note reverse order (NOT lat/
        zoom: 2 //default zoom level
    });
    ADD MAP CONTROLS AND LIMITS HERE


  </script>
```

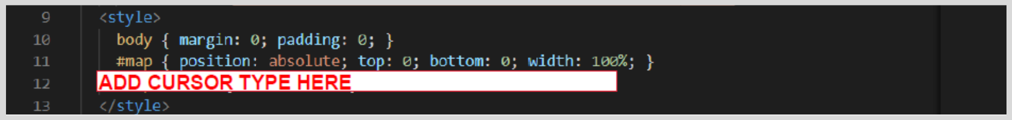
7a Add some **map controls** for ease of use. You'll add Zoom tools (+ and -), a fullscreen button, disable tile control (so the map stays flat), and disable scroll zoom (so the viewer can only zoom in and out with the buttons).

Add the following text after the close of your *var map* block, as shown above.

> *map.addControl(new mapboxgl.NavigationControl());*
> *map.dragRotate.disable();*
> *map.touchZoomRotate.disableRotation();*
> *map.scrollZoom.disable();*
> *map.addControl(new mapboxgl.FullscreenControl());*

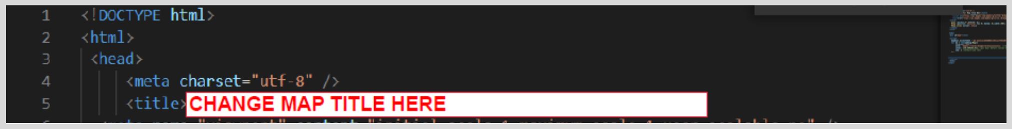
7b Now, to make it clear that you can pan around the map by clicking and dragging, change the **cursor type** to "move" (the crossed arrows) by adding the following in your <*style*> block:

> *#map canvas {cursor: move;}*



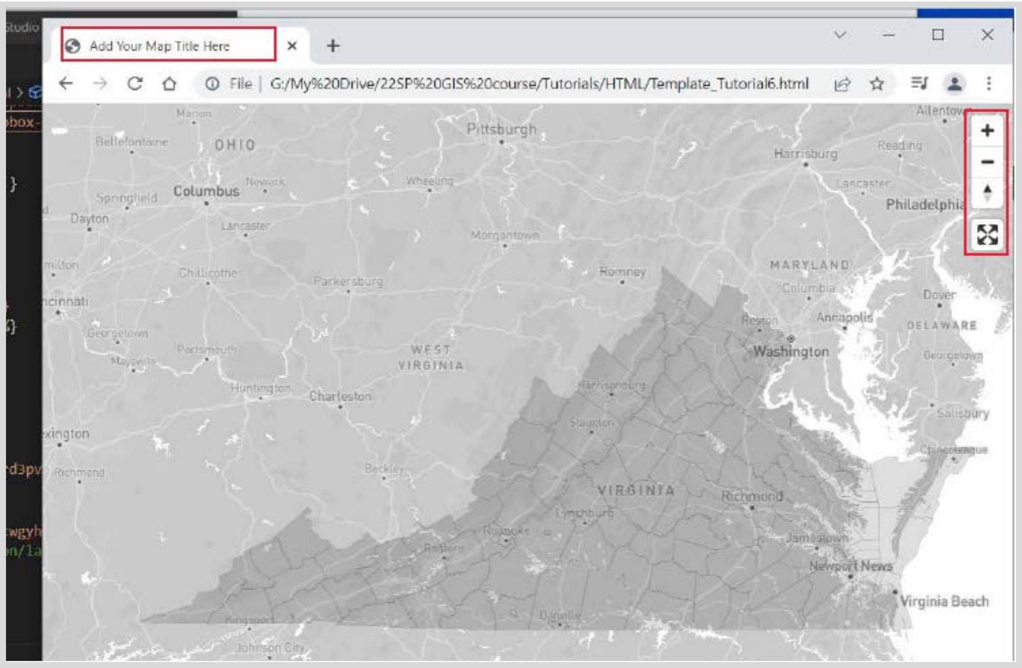
```
  <style>
    body { margin: 0; padding: 0; }
    #map { position: absolute; top: 0; bottom: 0; width: 100%; }
  ADD CURSOR TYPE HERE
  </style>
```

7c Lastly, change your **map title**:



```
<!DOCTYPE html>
<html>
 <head>
    <meta charset="utf-8" />
    <title>CHANGE MAP TITLE HERE
```

7d Now check out the changes in your Chrome map. Pan and zoom around to try out the new controls.



Add Your Map Title Here

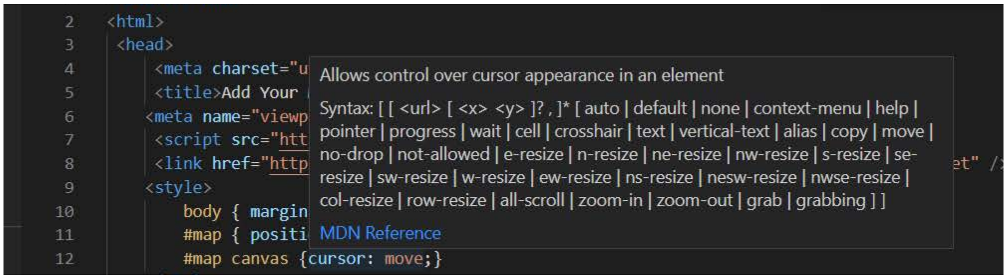
File | G:/My%20Drive/22SP%20GIS%20course/Tutorials/HTML/Template_Tutorial6.html

GIS for Designers
VT | A+D | SU22

– Bonus –

Step 8: Try out some different cursor types and map control options.

8a Hover over the cursor section of your code in Visual Studio to see some different options:



```
<html>
 <head>
    <meta charset="u
    <title>Add Your
   <meta name="viewp
    <script src="htt
    <link href="http                                                  et" /
   <style>
       body { margin
       #map { positi
       #map canvas {cursor: move;}
```

Allows control over cursor appearance in an element

Syntax: [[<url> [<x> <y>]? ,]* [auto | default | none | context-menu | help | pointer | progress | wait | cell | crosshair | text | vertical-text | alias | copy | move | no-drop | not-allowed | e-resize | n-resize | ne-resize | nw-resize | s-resize | se-resize | sw-resize | w-resize | ew-resize | ns-resize | nesw-resize | nwse-resize | col-resize | row-resize | all-scroll | zoom-in | zoom-out | grab | grabbing]]

MDN Reference

8b Visual Studio won't be able to show you the map control options, since these are specific to Mapbox's language. However, you can see the possibilities in Mapbox's documentation page:

https://docs.mapbox.com/mapbox-gl-js/api/markers/#navigationcontrol
and
https://docs.mapbox.com/mapbox-gl-js/api/handlers/