

Exercise: Modify an existing web map app

You work as a technical professional for a large auto parts company based out of Sandusky, Ohio. Your supervisor has an idea for upgrading the corporate web app to include a map that geographically displays the company's distributor locations. You have been tasked with putting together a quick web mapping app prototype. The prototype will display a map centered on the location of the company headquarters. To build the prototype, you will use HTML, CSS, and JavaScript referencing the ArcGIS API for JavaScript.

Estimated completion time: 30 minutes

- Step 1: Locate a sample

The ArcGIS API for JavaScript online developer help includes a guide for getting started, an API reference describing the individual objects, and sample codeso that you can begin writing your app.

In a browser, go to the ArcGIS API for JavaScript online developer help:
<https://developers.arcgis.com/javascript>.

The ArcGIS API for JavaScript online developer help displays options for learning more about the API in your browser. For the purposes of this task, you are interested in sample code that demonstrates how to create a basic app displaying a map.

Click the Sample Code tab to view the available samples.

A table of contents for locating the samples by type is displayed in your browser. You will now use the table of contents to locate a viable sample. In this case, the prototype requires only a map.

In the table of contents, expand Mapping And Views.

Numerous samples related to mapping and views are listed in the table of contents. For the prototype, you will display only a basic 2D map.

At the top of the list, click Intro To MapView (2D) to view this sample.

- Step 2: View and verify the sample

Now that you have found a sample of a web mapping app, you will view the sample to verify that its capabilities match your current prototype requirements.

On the Intro To MapView - Create A 2D Map page, click View Live.

A web mapping app opens in your browser and displays a map that is geographically centered in Europe.

Click your browser's back button to return to the Intro To MapView sample page.

The sample page provides a step-by-step tutorial for creating an app similar to the sample. Your company's headquarters is located in Sandusky, Ohio. You will find out whether this sample will let you change the displayed location of the map.

Read through the tutorial steps (specifically the code snippets) to locate any information or code discussing or demonstrating changing the map display's extent.



What are the current coordinates used to center the MapView?

This sample meets the requirements for the requested prototype. The sample demonstrates how to display a map and how to center the map to a specific location. You must now modify the code to meet the requirements.

- Step 3: Modify the sample code

The Intro To MapView (2D) sample works perfectly for your prototype. You will access the code and modify a few items to make it specific to your company's needs. One existing problem is that you do not have any web editing software. A long process exists for allowing software to be installed on your computer, and the prototype is due tomorrow. How are you supposed to modify the sample code and make the completed code available to your supervisor?

Below the map, click Explore In The Sandbox.

The code used to build the sample and a sample preview are displayed in the sandbox. The sandbox allows you to modify and run the code to see the results in the same browser window. Coding in the sandbox is both intuitive and relatively uncomplicated. If you code something that breaks the application, do not worry; you can revert to its original state by refreshing the browser.

One change that you need to make to the web mapping app is to center the map to display Sandusky, Ohio.

Modify the MapView code to center the map at the following location:

X: **-82.712174**

Y: **41.429967**

At the top right of the page, click Refresh to view the changes made to the web mapping application.

You can see that the map is centered correctly, but it displays a large portion of North America. You will now set the initial zoom level for the map.

Modify the zoom parameter located within the MapView so the initial map displays Sandusky, Ohio.

Experiment with typing in a few integer values and refresh the code to view the results.

```
var view = new MapView({  
  container: "viewDiv",  
  map: map,  
  zoom: 13,  
  center: [-82.712174, 41.429967]  
});
```

Step 3a: Modify the sample code.



What integer value could be appropriate for the zoom level?

The modified sample is beginning to look like a great prototype. Another change that you should make is to modify the MapView code so the map does not fill the entire web page.



Where in the code would you change the height and width of the map?

Change both height and width declaration values to **500px**.

Click Refresh to view the changes.

Although the map no longer fills the entire web page, it would look better centered horizontally in the page.

You will now modify the #viewDiv style to horizontally center the map on the page.

Create two new rules for the margin-left and margin-right declaration properties, setting a declaration value of **auto** for both.

Ensure that both the padding and margin declaration properties are removed from the style.

```

<!DOCTYPE html>
<html>
<head>
  <meta charset="utf-8">
  <meta name="viewport" content="initial-scale=1,maximum-scale=1,user-scalable=no">
  <title>Get started with MapView - Create a 2D map - 4.0</title>
  <style>
    html,
    body,
    #viewDiv {
      margin-left: auto;
      margin-right: auto;
      height: 500px;
      width: 500px;
    }
  </style>

  <link rel="stylesheet" href="https://js.arcgis.com/4.0/esri/css/main.css">
  <script src="https://js.arcgis.com/4.0/"></script>

  <script>
    require([
      "esri/Map",
      "esri/views/MapView",
      "dojo/domReady!"
    ], function(Map, MapView) {

      var map = new Map({
        basemap: "streets"
      });

      var view = new MapView({
        container: "viewDiv",
        map: map,
        zoom: 13,
        center: [-82.712174, 41.429967]
      });

    });
  </script>
</head>
<body>
  <div id="viewDiv"></div>
</body>
</html>

```

Step 3b: Modify the sample code.



If you add a border to a map displayed in a web page, will the map border be coded in the CSS or in JavaScript?

Click Refresh to view the modifications you made to the web page.

- Step 4: Download the sample code

Be aware that changes to code in the sandbox are not automatically saved. However, you can copy the code to your computer's clipboard, paste it into Notepad, and then save it locally as an HTML file. You can also download the code to your local computer by clicking Download.

At the top right of the page, click Download to download the code to your local computer.

Browse to the file system where your browser stores downloaded files and open the file.

Notice that the code is identical to the code you modified in the sandbox and that the file extension is HTML. You can share the file over the web in many different ways, including incorporating it into your organization's existing web application.

Close your browser.