Image from NASA: https://commons.wikimedia.org/wiki/File:Earth_Western_Hemisphere_transparent_background.png#filelinks

Client-Side Web GIS

ArcGIS Maps SDK for JavaScript

AmericaView
Empowering Earth Observation Education
AmericaView.org - Est. 2003

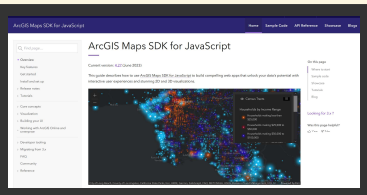
1

Background

2

Presenting Your Data

ArcGIS Maps SDK for JavaScript = offers fully customizable web maps and apps (requires coding)



https://developers.arcgis.com/javascript/latest/

API = Application Programming Interface

3

What is an API?

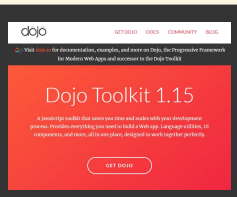
- API = Application Programming Interface
- URLs that provide code and functions that can be used to complete a task on the web
- We will explore the ArcGIS API for JavaScript in this course to make web maps

4

4

What is Dojo?

- A JavaScript toolkit/framework
- Saves time as you develop web apps
- Alternative to jQuery
- ArcGIS API for JavaScript makes use of Dojo

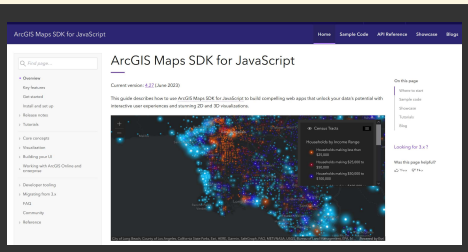


https://dojotoolkit.org/

5

5

Documentation



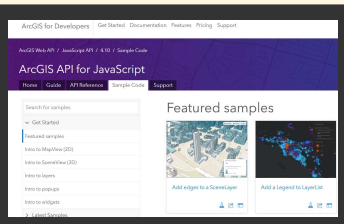
https://developers.arcgis.com/javascript/latest/

6

6

Sample Code

- Samples are provided on the API website: https://developers.arcgis.com/javascript/latest/sample-code/index.html
- Can start creating content by editing and combining the provided examples

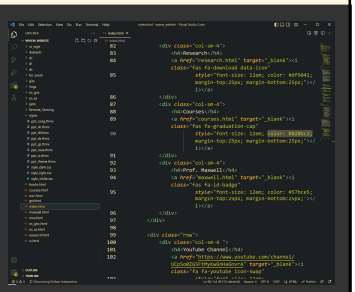


7

7

Using Text Editors

- Always use a plain text editor
- Don't type code in Word!
- Example: Notepad, Notepad++, Sublime Text, Atom, Brackets
- I like VS Code = https://code.visualstudio.com/

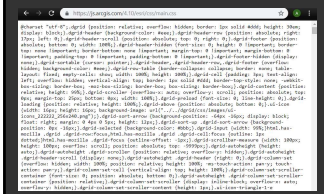


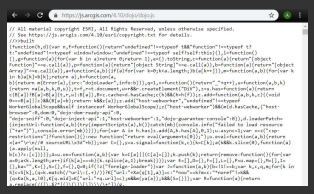
8

8

Referencing the API

```
<link rel="stylesheet" href="https://js.arcgis.com/4.27/esri/themes/light/main.css"/>
<script src="https://js.arcgis.com/4.27/"></script>
```





9

9

Connecting to Modules

- Calling modules provides access to specific functions made available by the API
- You must call modules that contains functions you need
- The modules called in the example are: Map, MapView, and FeatureLayer
- Modules will be called inside of a <script> tag and in a require function

```
require([
        "esri/Map",
        "esri/views/MapView",
        "esri/layers/FeatureLayer",
      ], function (Map, MapView, FeatureLayer)
{JS CODE GOES HERE}
```

10

10

Functions and Methods

- After calling in the modules, you will then call functions needed
- In the example, the Map, MapView, and FeatureLayer functions are being called
- Functions must be listed in the same order as the modules or the code will fail

```
require([
        "esri/Map",
        "esri/views/MapView",
        "esri/layers/FeatureLayer",
      ], function (Map, MapView, FeatureLayer)
{JS CODE GOES HERE}
```

11

11

Maps, Map Views, and Scene Views

12

12

Maps vs. Scenes

- Map = 2D projected map (generally uses Web Mercator)
- Scene = 3D scene of a 3D globe

Image from ArcGIS API for JavaScript

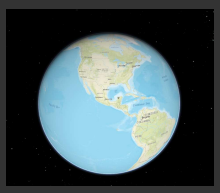


Image from ArcGIS API for JavaScript

13

13

Maps

- Class that contains properties and methods for storing, managing, and overlaying layers
- A map is rendered using a MapView or SceneView
- Name the Map
- Define a base map
- Define elevation surface
- Define operational layers

```
const map = new Map({
        basemap: "hybrid",
    });
```

https://developers.arcgis.com/javascript/latest/api-reference/esri-Map.html

14

14

Basemaps

- Available base maps

https://developers.arcgis.com/javascript/latest/api-reference/esri-Map.html

Value	Source
topo	https://services.arcgisonline.com/ArcGIS/rest/services/World_Topo_Map/MapServer
streets	https://services.arcgisonline.com/ArcGIS/rest/services/World_Street_Map/MapServer
satellite	https://services.arcgisonline.com/ArcGIS/rest/services/World_Imagery/MapServer
hybrid	https://services.arcgisonline.com/ArcGIS/rest/services/Reference/World_Boundaries_and_Places/MapServer https://services.arcgisonline.com/ArcGIS/rest/services/World_Imagery/MapServer
dark-gray	https://services.arcgisonline.com/ArcGIS/rest/services/Canvas/World_Dark_Gray_Reference/MapServer https://services.arcgisonline.com/ArcGIS/rest/services/Canvas/World_Dark_Gray_Base/MapServer
gray	https://services.arcgisonline.com/ArcGIS/rest/services/Canvas/World_Light_Gray_Reference/MapServer https://services.arcgisonline.com/ArcGIS/rest/services/Canvas/World_Light_Gray_Base/MapServer
national-geographic	https://services.arcgisonline.com/ArcGIS/rest/services/NatGeo_World_Map/MapServer
oceans	https://services.arcgisonline.com/arcgis/rest/services/Ocean/World_Ocean_Reference/MapServer https://services.arcgisonline.com/arcgis/rest/services/Ocean/World_Ocean_Base/MapServer
osm	OpenStreetMap [1,2,3,4,5]
terrain	https://services.arcgisonline.com/ArcGIS/rest/services/Reference/World_Reference_Overlay/MapServer https://services.arcgisonline.com/ArcGIS/rest/services/World_Terrain_Base/MapServer
dark-gray-vector	Dark Gray Canvas [v2]
gray-vector	Light Gray Canvas [v2]
streets-vector	World Street Map [v2]
streets-night-vector	World Street Map (Night) [v2]
streets-navigation-vector	World Navigation Map [v2]
topo-vector	https://services.arcgisonline.com/arcgis/rest/services/Elevation/World_Hillshade/MapServer World Topographic Map [v2]
streets-relief-vector	https://services.arcgisonline.com/arcgis/rest/services/Elevation/World_Hillshade/MapServer World Street Map (with Relief) [v2]

15

15

MapView

- Display 2D view of a map instance
- Define map to be rendered
- Define container (generally a div on the page)
- Can define map center
- Can define map zoom
- Can define extent as xmin, ymin, xmax, ymax
- Can define spatial reference

```
const view = new MapView({
  container: "viewDiv",
  map: map,

  extent: { // autocasts as new Extent()
    xmin: -9177811,
    ymin: 4247000,
    xmax: -9176791,
    ymax: 4247784,
    spatialReference: 102100
  }
});
```

https://developers.arcgis.com/javascript/latest/api-reference/esri-views-MapView.html 16

16

MapView

- Display 2D view of a map instance
- Define map to be rendered
- Define container (generally a div on the page)
- Can define map center
- Can define map zoom
- Can define extent as xmin, ymin, xmax, ymax
- Can define spatial reference

```
const map = new Map({
    basemap: "hybrid",
  });

  const view = new MapView({
    container: "viewDiv",
    map: map,
    center: [-80.5, 38.5],
    zoom: 8,
  });
```

https://developers.arcgis.com/javascript/latest/api-reference/esri-views-MapView.html 17

17

Place Map View on Page

- Most reference a container (div)
- Include CSS to set up the div
- Reference div in HTML <body>

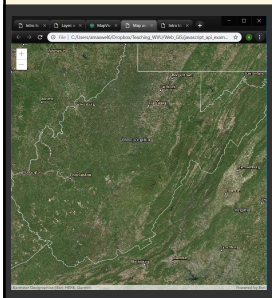
```
<body>
  <div id="viewDiv"></div>
</body>
```

```
<style>
    html,
    body,
    #viewDiv {
      padding: 0;
      margin: 0;
      height: 100%;
      width: 100%;
    }
  </style>
```

https://developers.arcgis.com/javascript/latest/api-reference/esri-views-MapView.html 18

18

MapView



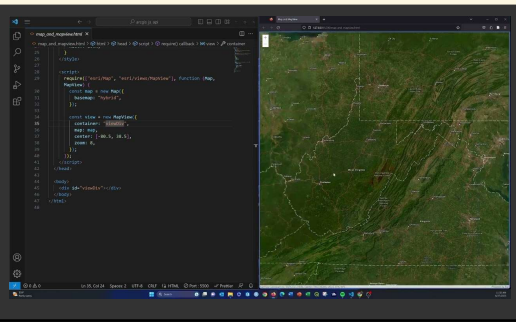
```
const map = new Map({
        basemap: "hybrid",
      });

      const view = new MapView({
        container: "viewDiv",
        map: map,
        center: [-80.5, 38.5],
        zoom: 8,
      });
```

https://developers.arcgis.com/javascript/latest/api-reference/esri-views-MapView.html

19

19

Video: Maps and Map Views



20

20

SceneView

- Allows map to be viewed on a 3D globe as opposed to a 2D projected map space
- Must define container
- Must define map to be rendered
- Define center
- Define scale or zoom
- Define source of elevation data with map

```
<script>
      require(["esri/Map",
"esri/views/SceneView"], function (Map,
SceneView) {
        const map = new Map({
          basemap: "terrain",
          ground: "world-elevation",
        });

        const view = new SceneView({
          container: "viewDiv",
          map: map,
          center: [-80.5, 38.5],
          scale: 100000,
        });
      });
    </script>
```

https://developers.arcgis.com/javascript/latest/api-reference/esri-views-SceneView.html

21

21

Defining Camera Position

- Define position of camera using:
  - Position
    - Latitude coordinate
    - Longitude coordinate
    - Elevation
  - Heading (0-360)
  - Tilt (0-90)

```
const map = new Map({
    basemap: "satellite",
    ground: "world-elevation",
  });

  const view = new SceneView({
    container: "viewDiv",
    map: map,
    camera: {
      position: [
        -100, // lon
        47.8, // lat
        2000, // elevation in meters
      ],

      heading: 0,
      tilt: 90,
    },
  });
```

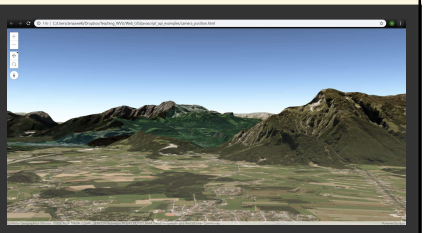
https://developers.arcgis.com/javascript/latest/api-reference/esri-views-SceneView.html

22

22

Defining Camera Position

```
const view = new SceneView({
        container: "viewDiv",
        map: map,
        camera: {
          position: [
            -13.04,
            47.8,
            1000,
          ],

          heading: 180,
          tilt: 90,
        },
      });
```

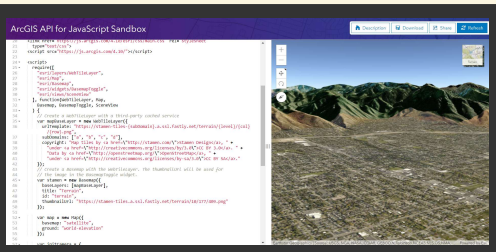


https://developers.arcgis.com/javascript/latest/api-reference/esri-views-SceneView.html

23

23

Custom Base map

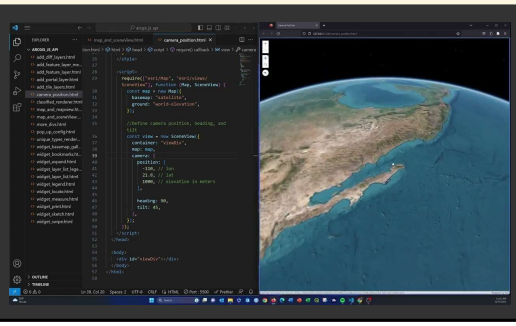


https://developers.arcgis.com/javascript/latest/sample-code/basemap-custom/index.html

24

24

Video: Maps and Scene Views



25

25

Adding Layers

26

26

Add Feature Layer

- Must call FeatureLayer module
- Must call FeatureLayer method
- Use map.add() to add to map

```
const countyTemp = new FeatureLayer({
  url: "URL",
});
map.add(countyTemp);
```

https://developers.arcgis.com/javascript/latest/api-reference/esri-layers-FeatureLayer.html

27

27

Add Feature Layer

- Can also add feature layer when defining the map using layers parameter

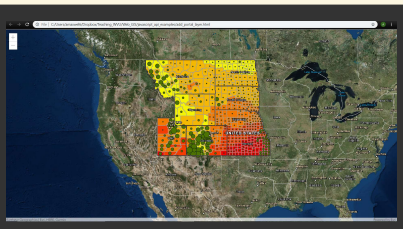
https://developers.arcgis.com/javascript/latest/api-reference/esri-layers-FeatureLayer.html

```
require([
        "esri/Map",
        "esri/views/MapView",
        "esri/layers/FeatureLayer",
      ], function (Map, MapView, FeatureLayer) {
          const countyTemp = new FeatureLayer({
            url: "URL",
          });

          //Define Feature Layer as Variable
          const stateBounds = new FeatureLayer({
            url: "URL",
          });

          //Add Feature Layers to Map
          const map = new Map({
            basemap: "hybrid",
            layers: [countyTemp, stateBounds],
          });
          const view = new MapView({
            container: "viewDiv",
            map: map,
            center: [-105, 45],
            zoom: 5,
          });
        });
```

28

28

Add Layer from Portal Item



https://developers.arcgis.com/javascript/latest/sample-code/layers-portal/index.html

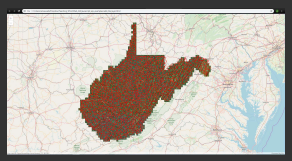
```
require([
        "esri/Map",
        "esri/views/MapView",
        "esri/layers/Layer",
      ], function (Map, MapView, Layer) {
        const map = new Map({
          basemap: "hybrid",
        });

        const view = new MapView({
          container: "viewDiv",
          map: map,
          center: [-106.5, 41],
          zoom: 5,
        });

        Layer.fromPortalItem({
          portalItem: {
            id: "ac427198c9bb463cadd2ac57456e0f8e",
          },
        }).then(function (layer) {
          map.add(layer);
        });
      });
```

29

29

Add Tile Layer

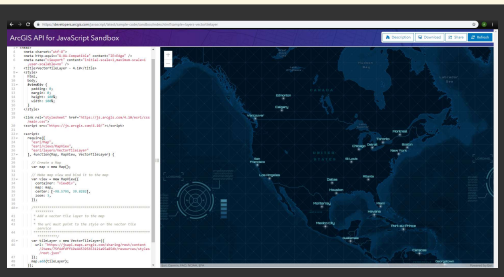
```
require([
        "esri/Map",
        "esri/views/MapView",
        "esri/layers/FeatureLayer",
        "esri/layers/TileLayer",
        "esri/layers/MapImageLayer",
        "esri/layers/VectorTileLayer",
      ], function (
        Map,
        MapView,
        FeatureLayer,
        TileLayer,
        MapImageLayer,
        VectorTileLayer
      ) {
        const naip = new TileLayer({
          url: "URL",
          format: "jpgpng",
        });
        const map = new Map({
          basemap: "osm",
          layers: [naip],
        });
        const view = new MapView({
          container: "viewDiv",
          map: map,
          center: [-80.5, 38.5],
          zoom: 8,
        });
      });
```



30

30

Add Vector Tile Layer

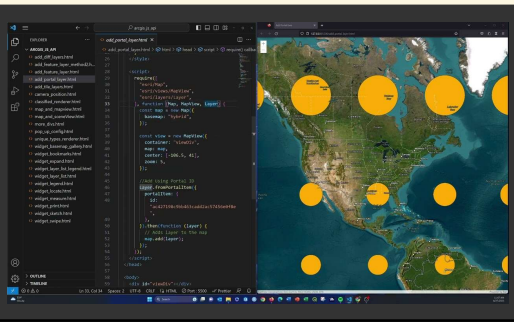


https://developers.arcgis.com/javascript/latest/sample-code/sandbox/index.html?sample=layers-vectortilelayer

31

31

Video: Adding Layers



32

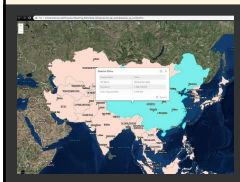
32

Pop-Ups and Renderers

33

33

Configure Pop-Ups



```
const featureLayer = new FeatureLayer({
  url: "URL",
  outFields: ["*"],
  popupTemplate: template,
});
map.add(featureLayer);
```

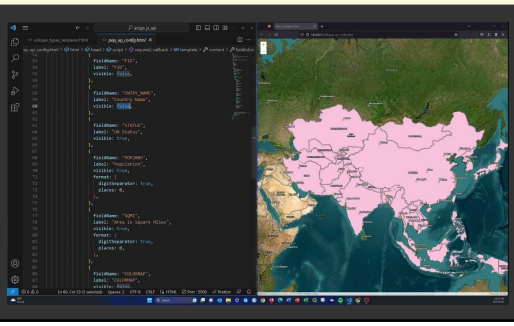
```
const template = {
    // autocasts as new PopupTemplate()
    title: "Data for {CNTRY_NAME}",
    content: [
      {
        type: "fields",
        fieldInfos: [
          {
            fieldName: "FID",
            label: "FID",
            visible: false,
          },
          {
            fieldName: "CNTRY_NAME",
            label: "Country Name",
            visible: false,
          },
          {
            fieldName: "STATUS",
            label: "UN Status",
            visible: true,
          },
        ],
      },
    ],
  };
```

https://developers.arcgis.com/javascript/latest/api-reference/esri-PopupTemplate.html

34

34

Video: Configure a Pop-Up



35

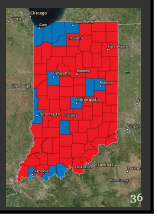
35

Use Renderers (Unique Types)

```
const rep = {
  type: "simple-fill",
  color: "#ff0000",
  style: "solid",
  outline: {
    width: 2,
    color: [0, 0, 0, 0.5],
  },
};
```

```
const voteRend = {
  type: "unique-value",
  legendOptions: {
    title: "2018 Presidential Election",
  },
  defaultSymbol: rep,
  defaultLabel: "Other",
  field: "winner08",
  uniqueValueInfos: [
    {
      value: "democrat",
      symbol: dem,
      label: "Democrat",
    },
    {
      value: "republican",
      symbol: rep,
      label: "Republican",
    },
  ],
};
```

```
const featureLayer = new FeatureLayer({
  url: "URL",
  renderer: voteRend,
});
map.add(featureLayer);
```



https://developers.arcgis.com/javascript/latest/api-reference/esri-renderers-Renderer.html

36

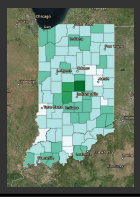
36

Use Renderers (Continuous with Class Breaks)

```
const 140g = {
  type: "simple-fill",
  color: "#b30000",
  style: "solid",
  outline: {
    width: 0.2,
    color: [0, 0, 0, 0.5],
  },
};
```

```
const renderer = {
  type: "class-breaks",
  field: "med_income",
  legendOptions: {
    title: "Median Income",
  },
  defaultSymbol: {
    type: "simple-fill",
    color: "black",
    style: "backward-diagonal",
    outline: {
      width: 0.5,
      color: [50, 50, 50, 0.6],
    },
  },
  defaultLabel: "no data",
  classBreakInfos: [
    {
      minValue: 0,
      maxValue: 39999,
      symbol: 140g,
      label: "< $40,000",
    }
```

```
const featureLayer = new FeatureLayer({
    url: "URL",
    renderer: renderer,
  });
map.add(featureLayer);
```



https://developers.arcgis.com/javascript/latest/api-reference/esri-renderers-Renderer.html

37

37

Types of Renderers

Visualization Type	Renderer
Location Only	SimpleRenderer, HeatmapRenderer
Unique Values	UniqueValueRenderer
Class Breaks	ClassBreaksRenderer
Continuous Color/Size	SimpleRenderer, UniqueValueRenderer, visualVariable
Multivariate	SimpleRenderer, UniqueValueRenderer, visualVariable

https://developers.arcgis.com/javascript/latest/api-reference/esri-renderers-Renderer.html

38

38

Defining Styles

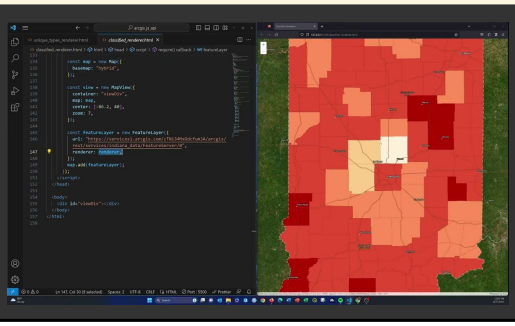
Geometry Type	Symbol Types
Point	SimpleMarkerSymbol, PictureMarkerSymbol, TextSymbol
Polyline	SimpleLineSymbol, TextSymbol
Polygon	SimpleFillSymbol, PictureFillSymbol, SimpleMarkerSymbol, TextSymbol

https://developers.arcgis.com/javascript/latest/api-reference/esri-symbols-Symbol.html

39

39

Video: Working With Renderers



40

40

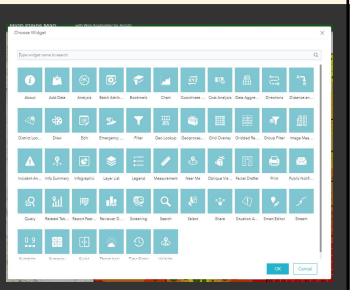
Widgets

41

41

Widgets

- Add functionality to your map
- Do not require coding
- Common widgets:
  - Basemap Gallery
  - Bookmark
  - Draw
  - Edit
  - Layer List
  - Legend
  - Measure
  - Print
  - Query
  - Select



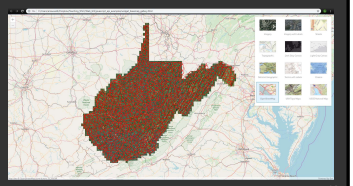
42

42

Basemap Gallery

```
require([
        "esri/Map",
        "esri/views/MapView",
        "esri/layers/FeatureLayer",
        "esri/layers/TileLayer",
        "esri/widgets/BasemapGallery",
      ], function (Map, MapView, FeatureLayer, TileLayer, BasemapGallery) {JS CODE GOES HERE}
```

```
const basemapGallery = new BasemapGallery({
          view: view,
        });
        view.ui.add(basemapGallery, {
          position: "top-right",
        });
```



https://developers.arcgis.com/javascript/latest/api-reference/esri-widgets-BasemapGallery.html

43

43

Bookmark Widget

```
require([
        "esri/Map",
        "esri/views/MapView",
        "esri/layers/FeatureLayer",
        "esri/layers/TileLayer",
        "esri/widgets/Bookmarks",
      ], function (Map, MapView, FeatureLayer, TileLayer, Bookmarks) {JS CODE GOES HERE}
```

```
const bookmarks = new Bookmarks({
  view: view,
});
view.ui.add(bookmarks, {
  position: "top-right",
});
```

https://developers.arcgis.com/javascript/latest/api-reference/esri-widgets-Bookmarks.html

44

44

Locate Button Widget

```
require([
        "esri/Map",
        "esri/views/MapView",
        "esri/layers/FeatureLayer",
        "esri/layers/TileLayer",
        "esri/widgets/Locate",
      ], function (Map, MapView, FeatureLayer, TileLayer, Locate) {JS CODE GOES HERE}
```

```
var locateBtn = new Locate({
          view: view,
        });

view.ui.add(locateBtn, {
          position: "top-left",
        });
```

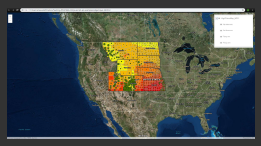
https://developers.arcgis.com/javascript/latest/api-reference/esri-widgets-Locate.html

45

45

Layer List Widget

```
require([
        "esri/Map",
        "esri/views/MapView",
        "esri/layers/Layer",
        "esri/widgets/LayerList",
      ], function (Map, MapView, Layer,
LayerList) {JS CODE GOES HERE}


view.when(function () {
          const layerList = new LayerList({
            view: view,
          });

          view.ui.add(layerList, "top-right");
        });
```



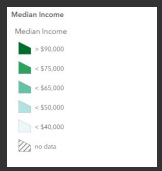
https://developers.arcgis.com/javascript/latest/api-reference/esri-widgets-LayerList.html

46

46

Legend Widget

```
require([
        "esri/Map",
        "esri/views/MapView",
        "esri/layers/FeatureLayer",
        "esri/widgets/Legend",
      ], function (Map, MapView, FeatureLayer, Legend) {JS CODE GOES HERE}

view.when(function () {
         const legend = new Legend({
           view: view,
           layerInfos: [
             {
               layer: featureLayer,
               title: "",
             },
           ],
         });

         view.ui.add(legend, "bottom-right");
       });
```



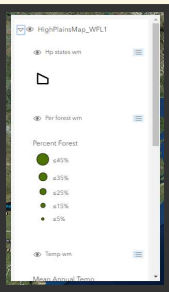
https://developers.arcgis.com/javascript/latest/api-reference/esri-widgets-Legend.html

47

47

Layer List + Legend Widget

```
require([
        "esri/Map",
        "esri/views/MapView",
        "esri/layers/Layer",
        "esri/widgets/LayerList",
      ], function (Map, MapView, Layer, LayerList) {JS CODE GOES HERE}


const layerList = new LayerList({
          view: view,
          listItemCreatedFunction: function (event) {
            const item = event.item;
            if (item.layer.type != "group") {
              item.panel = {
                content: "legend",
                open: true,
              };
            }
          },
        });
view.ui.add(layerList, "top-right");
```



https://developers.arcgis.com/javascript/latest/api-reference/esri-widgets-LayerList.html

48

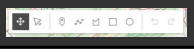
48

Sketch Widget

```
require([
        "esri/widgets/Sketch",
        "esri/Map",
        "esri/views/MapView",
        "esri/layers/FeatureLayer",
        "esri/layers/TileLayer",
        "esri/layers/GraphicsLayer",
        "esri/widgets/LayerList",
      ], function (
        Sketch,
        Map,
        MapView,
        FeatureLayer,
        TileLayer,
        GraphicsLayer,
        LayerList
      ) {JS CODE GOES HERE}
```

```
const layer = new GraphicsLayer({ title:
"Drawings" });

        const map = new Map({
          basemap: "osm",
          layers: [naip, featureLayer, layer],
        });

view.when(function () {
          const sketch = new Sketch({
            layer: layer,
            view: view,
          });

          view.ui.add(sketch, "bottom-right");
        });
```

https://developers.arcgis.com/javascript/latest/sample-code/sketch-geometries/index.html



49

49

Measurement Widget

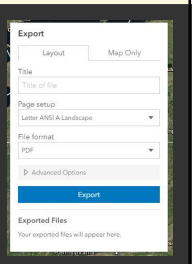
https://developers.arcgis.com/javascript/latest/sample-code/widgets-measurement-2d/index.html

50

50

Print Widget

```
require([
        "esri/Map",
        "esri/views/MapView",
        "esri/layers/FeatureLayer",
        "esri/widgets/Print",
      ], function (Map, MapView, FeatureLayer,
Print) {JS CODE GOES HERE}

view.when(function () {
          const print = new Print({
            view: view,
            // specify your own print service
            printServiceUrl:
              "URL",
          });
          view.ui.add(print, "top-right");
        });
```



https://developers.arcgis.com/javascript/latest/api-reference/esri-widgets-Print.html

51

51

Expand Widget

```
require([
        "esri/Map",
        "esri/views/MapView",
        "esri/layers/FeatureLayer",
        "esri/layers/TileLayer",
        "esri/widgets/LayerList",
        "esri/widgets/Expand",
        "esri/widgets/BasemapGallery",
      ], function (
        Map,
        MapView,
        FeatureLayer,
        TileLayer,
        LayerList,
        Expand,
        BasemapGallery
      ) {JS CDOE GOES HERE}
```

```
const llist = new LayerList({
  view: view,
  container: document.createElement("div"),
});

const bgExpand = new Expand({
  view: view,
  content: llist,
  expandIconClass: "esri-icon-zoom-out-fixed",
  collapseIconClass: "esri-icon-zoom-in-fixed",
});

view.ui.add(bgExpand, "top-right");
```

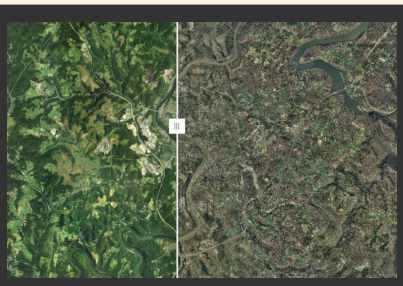
https://developers.arcgis.com/javascript/latest/api-reference/esri-widgets-Expand.html

52

52

Swipe Widget

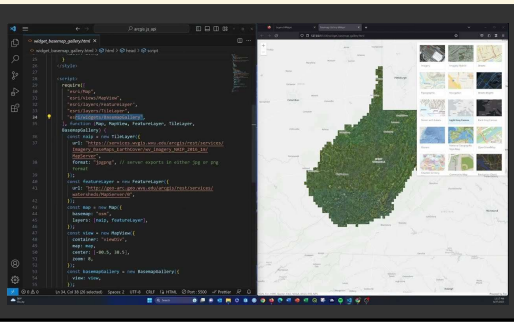
```
const swipe = new Swipe({
          leadingLayers: [naip],
          trailingLayers: [samb],
          position: 35,
          view: view,
        });
view.ui.add(swipe);
```



53

53

Video: Widgets



54

54

Adding Additional Content With Divs

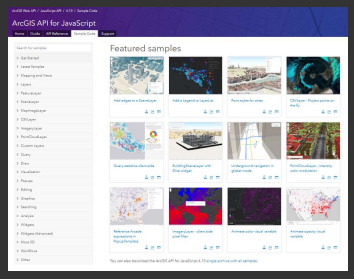
```
<style>
      html,
      body,
      #viewDiv {
        padding: 10px;
        margin: auto;
        height: 100%;
        width: 100%;
        max-width: 600px;
        max-height: 600px;
        background-color: #37363d;
      }

      .otherCont {
        padding: 10px;
        margin: auto;
        background-color: #37363d;
        min-height: 70px;
        color: #e3874f;
        text-align: center;
      }
    </style>
```

```
<body>
    <div class="otherCont">
      <h1>This is a header above the map.</h1>
      <p>This is a paragraph above the map.</p>
    </div>
    <div id="viewDiv"></div>
    <div class="otherCont">
      <p>This is a paragraph below the map.</p>
    </div>
  </body>
```

55

55

More Examples



https://developers.arcgis.com/javascript/latest/sample-code/index.html

56

56

This is the end of this lecture module.

Please return to the West Virginia View Webpage for additional content.

57

57