

# JavaScript Data Types

## String, Number, Boolean, Array, Object, Null, Undefined

- JavaScript Strings store a series of characters.

```
var answer="He is called 'Tommy'";
```

- JavaScript Numbers can be written with, or without decimals.

```
var x=27.00;
```

- JavaScript Booleans can only have two values: true or false.

```
var x=true;
```

- JavaScript Arrays store lists of data

```
var cars=["Toyota", "Honda", "Chevy"];
```

- JavaScript Objects store name and value pairs

```
var person={firstname:"Tom", lastname:"Smith", id:3874};
```

- Undefined** is the value of a variable that has not been assigned a value. Variables can be emptied by setting the value to **null**;

- You can use the JavaScript function `typeof()` to find the type of a variable.

```
typeof(3.14);           returns number
```

# References & Resources

- ESRI's Road Map for Web Developers:  
<http://blogs.esri.com/esri/arcgis/2014/02/21/esris-roadmap-for-web-developers/>
- Dave Bouwman's blog post "Transitioning to JavaScript":  
<http://blog.davebouwman.com/2014/03/03/transitioning-to-javascript/>
- Code Academy: <http://www.codecademy.com/tracks/javascript>
- Mozilla Developer Network JavaScript Guide:  
<https://developer.mozilla.org/en-US/docs/Web/JavaScript/Guide>
- Eloquent JavaScript: <http://eloquentjavascript.net/contents.html>

# Populate the Drop Down box

- A drop down box option is created using the syntax `box[option position] = new Option[option text, option value]`
- In this code, `box.length` grows by 1 each time a new option is added, so `box[box.length]` is always equal to the last option index.
- The variable `i` also grows by 1 each time the loop runs, so it can be used to reference each member of the `images` object in order.

```
function PageLoad() {  
    var box = document.getElementById("PhotoSelectBox");  
    for (var i in images) {  
        box[box.length] = new Option(images[i].title, i);  
    }  
}  
PageLoad();
```

# Load the Selected Photo

Type this code right before the `PageLoad();` function:

```
function MoveSelectedPhoto(newPhotoIndex) {  
    UpdateSelectedPhoto(newPhotoIndex);  
}  
  
function UpdateSelectedPhoto(newPhotoIndex) {  
    var photo = document.getElementById("Photo");  
    photo.src = images[newPhotoIndex].path;  
}  
  
document.getElementById("PhotoSelectBox").onchange =  
    function () { MoveSelectedPhoto(this.value); }
```

# Change Photo Dimensions

Type this code inside the  
UpdateSelectedPhoto(newPhotoIndex) function:

```
if (newPhotoIndex == 8) {  
    photo.width = "510";  
    photo.height = "737";  
} else {  
    photo.width = "737";  
    photo.height = "510";  
}
```

# Move First and Last

Type this code after the `UpdateSelectedPhoto(newPhotoIndex)` function:

```
function MoveFirst() { MoveSelectedPhoto(0); }  
function MoveLast() {  
    MoveSelectedPhoto(Object.keys(images).length - 1);  
}
```

Type this code after the `("PhotoSelectBox").onchange` event handler:

```
document.getElementById("FirstButton").onclick =  
function () { MoveFirst(); }
```

```
document.getElementById("LastButton").onclick =  
function () { MoveLast(); }
```

# Move Forward and Back

```
function MovePrevious() {  
    var box = document.getElementById("PhotoSelectBox");  
    var oldPhotoIndex = box.selectedIndex;  
    MoveSelectedPhoto(oldPhotoIndex - 1);  
}  
  
function MoveNext() {  
    var box = document.getElementById("PhotoSelectBox");  
    var oldPhotoIndex = box.selectedIndex;  
    MoveSelectedPhoto(oldPhotoIndex + 1);  
}  
  
document.getElementById("NextButton").onclick =  
function () { MoveNext(); }  
document.getElementById("PreviousButton").onclick =  
function () { MovePrevious(); }
```

# Change Selected Item in Drop Down

Type this code inside the

UpdateSelectedPhoto(newPhotoIndex) function:

```
var box = document.getElementById("PhotoSelectBox");  
box.selectedIndex = newPhotoIndex;
```



# Enable Navigation Buttons

Type this code below the MoveNext() function:

```
function EnableNavigationButtons() {  
    var firstButton = document.getElementById("FirstButton");  
  
    var previousButton =  
document.getElementById("PreviousButton");  
  
    var nextButton = document.getElementById("NextButton");  
  
    var lastButton =  
document.getElementById("LastButton");  
  
    firstButton.disabled = false;  
    previousButton.disabled = false;  
    nextButton.disabled = false;  
    lastButton.disabled = false;  
}
```

# Selectively Disable Navigation Buttons

Type this code inside the `EnableNavigationButtons()` function:

```
var box = document.getElementById("PhotoSelectBox");
var photoIndex = box.selectedIndex;

if (photoIndex == 0) {
    firstButton.disabled = true;
    previousButton.disabled = true;
} else if (photoIndex == Object.keys(images).length - 1) {
    nextButton.disabled = true;
    lastButton.disabled = true;
}
```

Type this code inside the `UpdateSelectedPhoto(newPhotoIndex)` function:

```
EnableNavigationButtons();
```

# Make a Slideshow!

Type this code after the `EnableNavigationButtons()` function:

```
var slideIndex = 0;
function PlaySlideshow() {
    MoveSelectedPhoto(slideIndex);
    slideIndex++;
    if (slideIndex >= Object.keys(images).length) {
        slideIndex = 0;
        return;
    }
    setTimeout(function () { PlaySlideshow(); }, 2000);
}

document.getElementById("SlideshowButton").onclick =
    function () { PlaySlideshow(); }
```

# Add require statement

---

At the very top of your .js put these opening statements

```
require([  
  "dojo/on",  
  Insert the paths to ESRI modules here  
  "dojo/domReady!" ],  
function (  
  on,  
  Insert variable names for ESRI modules here  
) {
```

Remember you don't need a variable name for dojo/domReady!

At very the bottom of your .js put this closing tag

```
});
```

paths to ESRI modules:

```
"esri/map",  
"esri/symbols/PictureMarkerSymbol",  
"esri/graphic",  
"esri/geometry/Point",  
"esri/InfoTemplate",
```

variable names for ESRI modules:

```
Map,  
PictureMarkerSymbol,  
Graphic,  
Point,  
InfoTemplate
```

# Completed ESRI API Code

---

```
var options = {  
    center: [-97.742581, 30.2837352],  
    zoom: 12,  
    basemap: "topo"  
};  
  
var map = new Map("mapDiv", options);  
  
on(map, "load", function () { loadGraphics(); PageLoad(); });  
  
var symbol = new PictureMarkerSymbol("Images/camera-icon.png",  
20, 20);
```

---

# Completed ESRI API Code

---

```
function loadGraphics() {
    for (var i in images) {
        var geometry = new Point(images[i].location);
        var attr = { "index": i, "Image": images[i].title };
        var infoTemplate = new InfoTemplate("Photos",
        "${Image}");
        var graphic = new Graphic(geometry, symbol, attr,
        infoTemplate);
        map.graphics.add(graphic);
    }

    on(map.graphics, "click", function (e) {
        UpdateSelectedPhoto(e.graphic.attributes.index);
    });
}

function zoomToGraphic(point) {
    map.centerAndZoom(point, 16);
}
```

---

# GIS Resources

---

ESRI JavaScript API Reference Guide: <http://js.arcgis.com>

Licensing:

<https://developers.arcgis.com/javascript/latest/guide/licensing/index.html>

ArcGIS Symbol Playground:

<https://developers.arcgis.com/javascript/3/samples/playground/index.html>

Collection of Symbols:

[https://developers.arcgis.com/javascript/3/samples/portal\\_symbols/index.html](https://developers.arcgis.com/javascript/3/samples/portal_symbols/index.html)

Dojo Reference Guide: <https://dojotoolkit.org/>

---

ATX GIS Day: <http://atxgisday.org/>

CenTexGIS: <http://centexgis.com/>

SWIGGIS: <http://www.swiggis.org/>

---

Website Source Code: <https://github.com/AubreyRhea/swiggis/>

Instructor Contact: Aubrey Drescher aubrhea@gmail.com

---