Google Maps

- Various APIs
  - Maps: types of maps, scale, center
  - Overlay objects: markers, polylines
  - Events: on map and overlay objects

- Check this website for an example on clicking event.
  - https://developers.google.com/maps/documentation/javascript/examples/infowindow-simple
- Other events: mouseover, mouseout, etc.

JavaScript:
Array, Loop, Data File

JavaScript Basics

- Statements
- Variables
- Events
- Functions

Statements

```
<script type="text/javascript">


var a = 10;

var b = 11;

var c;

c = a + b;

alert(‘The answer is’ + c);


</script>
```

JavaScript Variables

- Case sensitive
- Must begin with a letter or the underscore character
- Need to be declared
  - var variableName

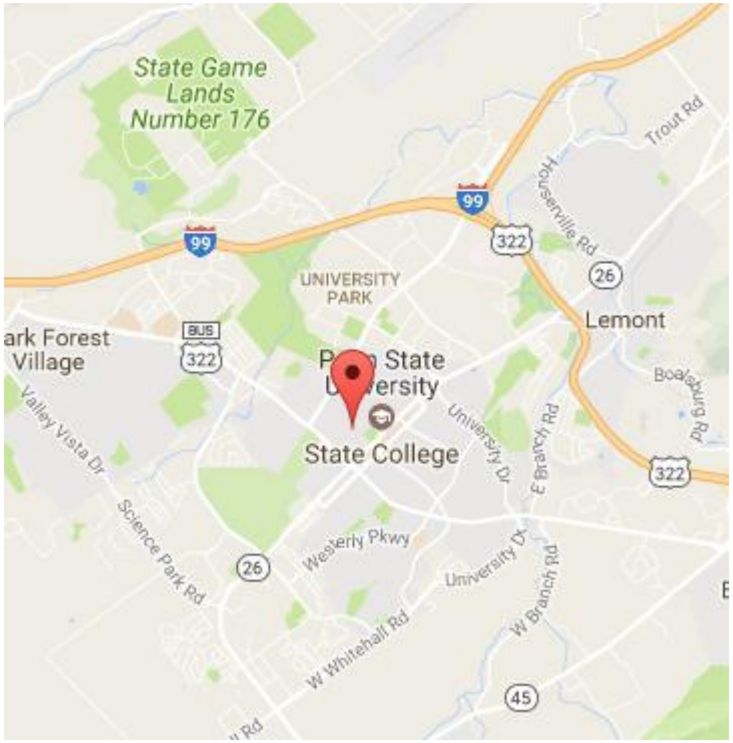
JavaScript Operators

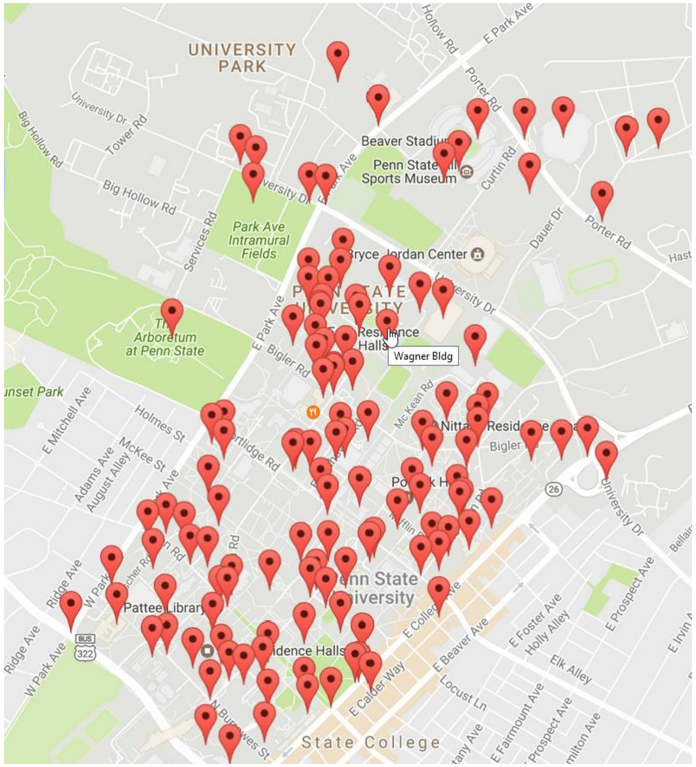
- Arithmetic Operators
  - +,-,*,/,%,++,--
- Assignment Operators
  - =, +=, -=, *=, /=, %=
- Comparison Operators
  - ==: value only
  - ===: value and type
  - !=
  - >,<,>=,<=
- Logical Operators
  - &&, ||, !

Functions

```
<script>
  function myFunction() {
    alert(Date());
  }
</script>

<button onclick="myFunction()">The Time is?</button>
```


State Game Lands Number 176
99
322
UNIVERSITY PARK
ark Forest Village
BUS 322
Penn State University
State College
Lemont
26
Housersville Rd
Trout Rd
Boalsburg Rd
University Dr
E Branch Rd
Valley Vista Dr
Science Park Rd
Westerly Pkwy
W Whitehall Rd
W Branch Rd
45


UNIVERSITY PARK
Beaver Stadium
Penn State All Sports Museum
Park Ave Intramural Fields
Bryce Jordan Center
PENN STATE UNIVERSITY
Residence Halls
Wagner Bldg
The Arboretum at Penn State
Sunset Park
Nittany Residence
Pollock Halls
Penn State University
Pattee Library
Residence Halls
State College
Big Hollow Rd
University Dr
E Park Ave
Porter Rd
Curtin Rd
Holmes St
Bigler Rd
E College Ave
E Beaver Ave
W Park Ave
Ridge Ave
E Foster Ave
Locust Ln
E Prospect Ave
Elk Alley
N Burrowes St

One Marker

```
function initMap() {
  // The location of Uluru
  const uluru = { lat: 40.798, lng: -77.862 };

  // The map, centered at Uluru
  const map = new google.maps.Map(document.getElementById("map"), {
    zoom: 14,
    center: uluru,
  });

 // The marker, positioned at Uluru
  const marker = new google.maps.Marker({
    position: uluru,
    map: map,
  });

}
```

Multiple Markers?

```
function initMap() {
  // The location of Uluru
  const uluru = { lat: 40.798, lng: -77.862 };

  // The map, centered at Uluru
  const map = new google.maps.Map(document.getElementById("map"), {
    zoom: 14,
    center: uluru,
  });

 // Marker 1
  const position1 = {lat …};
  const marker1 = new google.maps.Marker({
    position: position1,
    map: map,
  });

 // Marker 2
  const position = {lat …};
  const marker1 = new google.maps.Marker({
    position: position2,
    map: map,
  });
.
.

}
```

Any Better Ways?

A Better Way to Show Multiple Markers:

Array + Loop

Array

- A data structure to hold a set of data with the same attributes
- Benefit of using array
    - Repetition becomes easier.

```
loop from the 1st item to the last item {
    create a marker;
}
```

Array in JavaScript

- Simple array: elements are a simple data type.

```
var aListofNumber = [1,2,3,4];
var aListofString = ["a", "b", "c","d"];
```

Array Reference

- Use an indexing number

ArrayName[*indexNumber*]

```
var aListofNumber = [1,2,3,4];
var aListofString = ["a", "b", "c","d"];
```

```
aListofNumber[0]
aListofString[3]
```

Array in JavaScript

- Object array: elements are objects, which have multiple data attributes (same or different types).

```
var aListofPeople = [{name: "John", age: 28},
                     {name: "Jane", age: 24},
                     {name: "Joe", age: 20}];
```

Objects are grouped by {}.

```
aListofPeople[1].name
aListofPeople[2].age
```

In the Case of Your Assignment

```
crimes = [
 {date: "9/1/2022", lat: 40.81, lng:-77.86, location: "Wagner Bld"},
 {date: "9/1/2022",lat: 40.79, lng: -77.86, location: "Reber Bld“},
                    .
                    .
                  ];
```

Array Properties

- Most useful one: length

```
arrayName.length
```

```
var aListofNumber = [1,2,3,4];
```

```
aListofNumber.length = 4;
```

Array Methods

- Some important methods
  - push(): adding a new element
  - pop(): removing the last element
  - indexOf(): finding the position of a given element
  - forEach(): calling a function for each element

Processing Data in An Array

- Loop
- The For loop

```
for (statement 1; statement 2; statement 3) {
    code block to be executed
}
```

```
for (var i=0; i<crimes.length; i++) {

        var marker = new google.maps.Marker({
          position: ???,
          map: map
         });

}
```

- http://www.w3schools.com/js/js_loop_for.asp

Exercise 1: Preparing Data for Markers

- Go to the website: http://www.personal.psu.edu/xuz14/330/Assignments/Assignment1/
- Use the data in JSArray.txt to define an array of objects in JavaScript called crimes. Add the array before the statement window.initMap = initMap;

```
var crimes = [
    {Date:"9/1/2016", Offenses: "Harassment", Location: "Wagner Bldg", lat: 40.805634, lng: -77.859307},
    {Date:"9/1/2016", Offenses: "Suspicious Activity - Other", Location: "Reber Bldg", lat: 40.793368, lng: -77.864351},
    {Date:"9/1/2016", Offenses: "Health and Safety/RFA-Request for Assistance", Location: "Food Science Bldg", lat: 40.804138, lng: -77.861936},
    .
    .
    .
    {Date:"9/15/2016", Offenses: "PSU-Rules and Regulations", Location: "Pollock Rd & Bigler Rd", lat: 40.802493, lng: -77.857859}
];
window.initMap = initMap;
```

- Inside the initMap() function, add the following statements at the end.

```
for (var i=0; i<crimes.length; i++) {
    console.log(crimes[i]);
};
```

- Use the console tool to check whether all data entries are read correctly.
    - the Console tab.

Exercise 2: Showing the Markers

- Define marker coordinates: use the statement to define the uluru variable in the example as your reference.
    - You need to take the values from the data entry.
        - E.g., crimes[i].lat
- Define a marker with the given marker location.

Exercise 3: Showing Campus Location when Mouse is Over a Marker.

- Go to this website to learn more about a marker: https://developers.google.com/maps/documentation/javascript/markers
  - For more technical details, check this website: https://developers.google.com/maps/documentation/javascript/reference/marker
- Add one property in your definition of the marker and take this attribute from the array: crimes[i].Location.
  - Take a look at this page if you have no idea what property to add https://developers.google.com/maps/documentation/javascript/examples/marker-simple

Data File

- Converting the raw data into array is cumbersome.
- Can we read the file directly?
  - Yes!
  - There are many ways to do that.
- Some JavaScript libraries can help us.
- D3.js is one of them.
- D3.js offers a simpler approach.
  - To read a csv file or a JSON file
    - d3.csv ("filename", function(data) {});
      - CSV: all data is string
    - d3.json ("filename", function(data) {});
      - JSON: smarter, different data types.
        - E.g., latitude and longitude data
- The data file should be on a server (could be the same server with the js file).

```
d3.csv("crimes.txt", function(data) {
    console.log(data);
});
```

For JSON file, you can convert the csv file to JSON with some online tools, such as http://www.csvjson.com/csv2json

```
d3.json("crimes.json", function(data) {
    console.log(data);
});
```

Pay attention to the difference in latitude and longitude data.

Need this line before calling d3.csv or d3.json :

```
<script src=“//d3js.org/d3.v3.min.js"> </script>
```

https://my.up.ist.psu.edu/xuz14/330/csv_to_json_byD3.html

Comments for JavaScript Codes

- Comments are important to understand and maintain codes.
- In your assignment, you must add some comments (1 point)

```
var uluru = {lat: 40.796466,lng: -77.8649379};
var map = new google.maps.Map(document.getElementById('map'), {
  zoom: 12,
  center: uluru
});
```

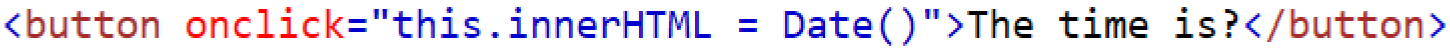
VS.

```
//the center of the map
var uluru = {lat: 40.796466,lng: -77.8649379};

//a new map with a given center and zooming level
var map = new google.maps.Map(document.getElementById('map'), {
  zoom: 12,
  center: uluru
});
```

JavaScript Events

- What is happening to HTML elements.
- User and browser events
    - onclick, onmouseover, onmouseout, onkeydown, etc.
- Example: onclick



```
<button onclick="this.innerHTML = Date()">The time is?</button>
```

- http://www.w3schools.com/js/tryit.asp?filename=tryjs_event_onclick

Bonus Points

- Purpose: encourage you to consider more interaction methods.
  - Mouseover, mouseclick, …

- Use an infowindow for each mouse over a marker.
  - Hint
    - Listen to the click event on a marker
    - Call a function to show an infowindow with appropriate information.

Exercise 4: Displaying More Details of A Reported Crime

- Add a listener to each marker for mouse over action.
    - Infowindow will be the object to add when moving the cursor over a marker. The content of the infowindow should be taken from the array.
- Search examples online (keywords: google map marker, infowindow).

Some Issues You May Encounter

- https://leewc.com/articles/google-maps-infowindow/