

Javascript Specialist Designation Program

Lab Exercise #15

A great feature on any application is the ability to grab someone's location. Even better, having the ability to get directions to that certain location.

Lab Instructions

1) Run your Brackets development environment, create a new file and quickly save it under the file name **fifteenth_javascript_lab.html**

2) Key in the basic HTML document structure and set **Lab #15** as the content for the title tag.

3) In the body of the HTML document add a heading 1 tag with the content, "Show location on Google Maps."

4) We're going to need a <button> for the user to find their location, and a <p> tag to link them to Google.

Start with the button, give it an id of "getLocationBtn" and the content of "Get Location".

Next, open a paragraph tag, give it an id of "mapsLink". We will leave the content empty since the coordinates have not been located yet.

```
<h1>Show location on Google Maps</h1>
<button id='getLocationBtn'> Get
Location</button>
<p id='mapsLink'></p>
```

5) Now for some Javascript. Create a separate Javascript file and quickly save it as **main.js**.

Link the HTML file to the Javascript file.

6) Open your **main.js** file and create the window.onload function.

Inside the onload function, we're going to add an event listener for the button. Search the document for an element with the id "getLocationBtn" then add a "click" event listener and run a function name "grabLocation".

```
window.onload = function() {
document.getElementById('getLocationBtn')
.addEventListener('click', grabLocation)
};
```

7) The response function "grabLocation" will be responsible for running the HTML5 Geolocator. Start with calling the navigator object, then calling the geolocation object, then invoking the getCurrentPosition function. The success callback will be named "showGoogleLink" and we will keep error as "error".

```
function grabLocation() {
navigator.geolocation.getCurrentPosition(
showGoogleLink, error);
};
```

8) After the grabLocation() function, we want to declare a variable called "myLocation" initialize it by searching the document for an element with the id "mapsLink"

```
var myLocation =
document.getElementById('mapsLink');
```

After, let's create a simple "error" function. Inside the error() function, give an alert() with a message "There was an error".

```
function error() {
alert('There was an error');
};
```

9) Lastly, let's create our success function which is named "showGoogleLink". The "showGoogleLink" will have a parameter named "position". Inside the function, we're going to set the innerHTML of the variable myLocation

equal to an anchor tag with an href. The href will be the URL for Google Map's search.

The URL starts with the usual "https://www.google.com/maps/@" then we will concatenate the latitude by calling the position object, then coords object and the latitude value. Concatenate a comma, and lastly concatenate and longitude value, the same way you did latitude.

Add a content of the anchor tag as "Click Me".

Your code should look something like this:

```
function showGoogleLink(position) {  
  
    myLocation.innerHTML = "<a href=https://www.google.com/maps/@" +  
    position.coords.latitude + ',' + position.coords.longitude + "> CLICK ME </a>"  
  
};
```

10) Click File Save to save the current version of your document. Navigate to the HTML document using your operating system and double click it. The document should open in your default browser.

Click "Get Location" and look at yourself through Google Maps!

Challenge Yourself

Enhance your skills by attempting the exercises below.

1) `getCurrentPosition()` accepts an optional [PositionOptions](#) object. Play around with it and see what it has to offer.

2) Put your site on the web. We've arranged a special deal with Blue Host. Visit <http://www.bluehost.com/track/learntoprogram/> and click "Get Stated Now." You will be able to access web hosting plans as low as \$3.49 a month. Once Blue Host takes you through the process of creating your domain and web server upload your lab and post the URL for the others in the class to see.