1. Just a reminder, when you save your pages make sure they are text, but that they have the `.html` extension and not `.txt`. As you modify your pages to add functionality, be sure to save separate copies.
2. Let's use Javascript to change the text size on the web-page. We will use the `select` form element, with three choices. There will be some text, in a div element, which will be modified. Once again, we can use `querySelector`. Here, it will modify all paragraph elements ***p***, so the parameter in `querySelector` will ***NOT*** have a ***#***. The style element modified will be `fontSize`. Following our normal design procedure, we render the form first, making sure we have the correct URL and form element. Once that is set, we need to:
    a. Name the `select` element.
    b. Put an event in the `select` element.
    c. Write the Javascript function in the `script` element.

    This give us (http://jcsites.juniata.edu/faculty/kruse/it105/inClass/js14.html):

```
<!DOCTYPE html>
<html lang=”en”>
   <head>
      <title>Hello!</title>
      <script>

        function changeSize()
        {

document.querySelector("p").style.fontSize=document
. querySelector("#sizeChange").value;
        }

      </script>
   </head>
   <body>
      <p>This is some text.</p>
      <form name=inclass>
        <select id="sizeChange"
onchange="changeSize();">
         <option value="large">Large Text</option>
```

```
          <option value="initial">Medium
Text</option>
          <option value="small">Small Text</option>
         </select>
       </form>
    </body>
```

3. There are two more built-in pop-up functions, `confirm` and `prompt`, which are similar to the `alert` function we're already familiar with. While `alert` merely presents something, `confirm` and `prompt`, receive input back from the user. `confirm` returns a Boolean (true or false) value to Javascript, while `prompt` returns a string value.
4. Here is a sample of Javascript code demonstrating the `confirm`, note that this code would be in the `script` element:

```
function confChk() {
 var areYouSure = confirm("Is this your fav.
class?");
 if (areYouSure) {
  alert("A in the course!");
 } else {
  alert("Well, I guess you like FYC better?");
 }
}
```

5. Here is a sample of Javascript code demonstrating the `prompt`, note that this code would be in the `script` element:

```
function promptChk() {
 var favDrink = prompt("Enter your favorite
drink");
 if (favDrink=="water") {
  alert("thirsty!");
 } else {
  alert("What about water?");
 }
}
```

6. Finally, the last built-in Javascript function we will consider is ***geolocation***. ***Note, this will not work in Chrome. Google requires that the page is from a secure site, as given by*** `https`. Google's explanation is: https://developers.google.com/web/updates/2016/04/geolocation-on-secure-contexts-only
   Let's proceed with the code: (http://jcsites.juniata.edu/faculty/kruse/it105/inClass/js15.html):

```
<!DOCTYPE html>
```

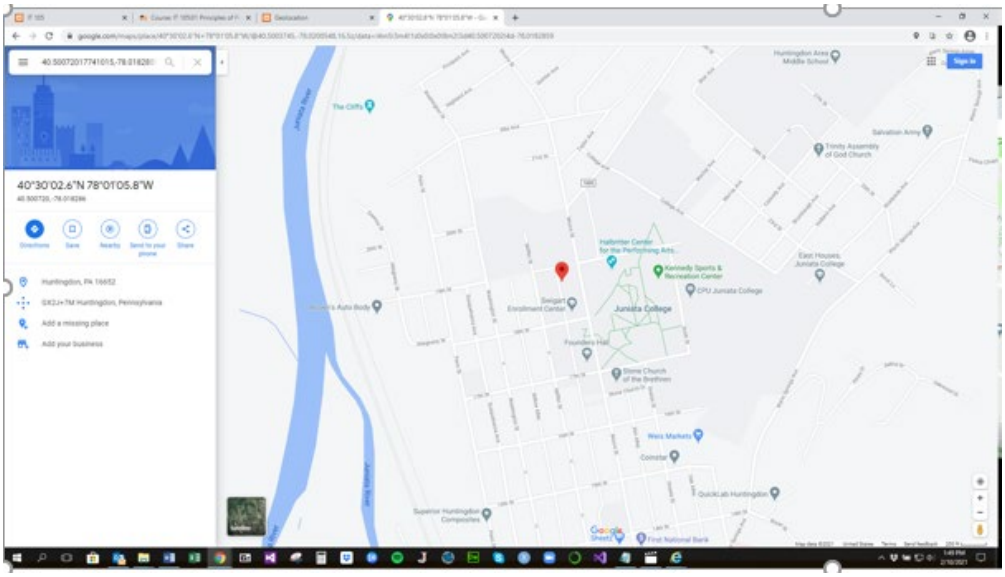
```
<html lang="en">
  <head>
    <title>Geolocation</title>
    <script>

navigator.geolocation.getCurrentPosition(function(p
osition) {
    document.write(position.coords.latitude+','+pos
ition.coords.longitude);
      });
    </script>
    </head>
    <body>
      Hello, world!
    </body>
</html>
```

7. `navigator.geolation.getCurrentPosition()` is a function, which get the current latitude and longitude of my computer. The argument to this function is actually another function, called a ***callback function***, which is the function which should be called after we've gotten the current position.
8. The callback function is highlighted, and the call becomes:
```
navigator.geolation.getCurrentPosition(function(pos
ition){
document.write(position.coords.latitude + ' ' +
position.coords.longitude); });
```
9. Note that `document.write` is like `querySelector`, it can access our HTML document, and it will write the position results to our page. Note, as discussed above, your browser (or network) might not allow code to make changes to your webpage. Here are some screen shots:

clicking "Allow once" then gives

40.50073309301963,-78.01828754676214

10. If we copy and paste this latitude and longitude into Google Maps we get:



which looks pretty close to my office in BAC…