**Programming Problem (50% -- 75 points):**

Create a web application which displays stock information from the included XML file. This project will involve the following: XML, CSS, JS/JQ, event programming, and DOMs (XML & HTML).

**The program works as follows:**

* It reads in an XML file (stocks.xml) which contains information about stocks. The XML is placed in an XML document object.
* Display the list of stock ticker symbols. When displayed, sort the stocks alphabetical order by ticker symbol. Note that these stocks are not sorted in the XML file. Sorting should be done sorting routine in JS/JQ. **IMPORTANT: Do not create a sorted list simply by rearranging the <Stock> elements in the XML file (as I will check for this).**
* When the mouse hovers over one of the stocks in the list, the HTML for the additional information about the stock is generated and placed in the <div id=”stockInfo”> element (see Figure 2). Note that this part involves accessing the XML through the DOM to get the additional information. In addition, the color of the selected item will change from navy to red.
* When the mouse no longer hovers over the item, the color of the text will change back to navy. Between 3-5 seconds must pass, during which time the information will linger. After that, the <div id=”stockInfo”> element must clear*.*

**Provided:**

* XML file (do not provide a JSON equivalent)
* HTML file

**What to do:**

* You must duplicate the functionality of the program (demonstrated in class and illustrated in the below figures).
* For testing purposes, another XML document (compliant with the same schema) may be substituted. While the sample XML file contains five (5) stocks, your solution will have to correctly process all the stocks in the test file.
* **In order for this to work, you will need to open this as a web site project in Visual Studio and run it using the IIS (with the HTML file as the designated start page). For full credit, your solution must work in the Chrome, Edge, and FF browsers.**
* Include a READ-ME file with your solution that documents any known bugs and indicates which browsers your solution has been successfully tested in.
* Zip your full solution and upload to the D2L dropbox for the midterm assignment.
* Partial credit will be awarded if some subset of the required functionality is working.

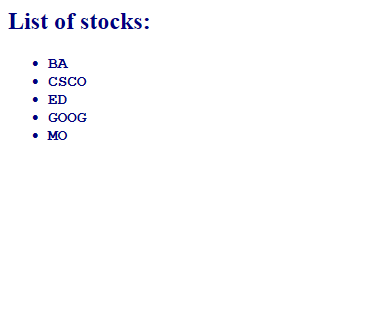


Figure 1 – Mouse Out



Figure 2 - Mouse Over