

Interactive Web Programming - Lab 1

JavaScript

This lab will expose you to XHTML forms and some basic JavaScript. Refer to notes posted on the class web page. As you work through this material, you will notice some commonalities between JavaScript and other languages we have studied (variables, functions, selection-if, repetition-loops).

JavaScript is embedded in XHTML files, and so you will write your code using a text editor, such as Notepad on a PC or Text Edit on a Mac. On Linux, you can use pico or nano or emacs or vi, etc..

1. **Modify your personal website.**

- 1.1. Create a new folder in your personal website folder called javascript.
- 1.2. Save the WebPages you create in sections 2 and 3 into that folder.
- 1.3. When you are done with sections 2 and 3 create links on your websites index.html page to the documents you created in sections 2 and 3.

2. **XHTML buttons and JavaScript**

2.1. [Look at the following HTML file:](#)

```
<html>
  <head>
    <script type="text/javascript">
      function show_alert()
      {
        alert("I am an alert box!");
      }
    </script>
  </head>
  <body>
    <p>
      <input type="button" onclick="show_alert()" value="click me" />
    </p>
  </body>
</html>
```

- 2.2. Use the example above to create a new document called jsbutton.html. In your document, create three buttons, with values "RED", "BLUE" and "GREEN" (instead of "click me").
- 2.3. Modify the onclick function call for each button so that when you click on each button, a window pops up and displays a message like "you clicked on the RED button" (or something more creative than that), instead of the message "I am an alert box". Whatever message you use, it must contain text that indicates which color button you clicked on and the message must be different for each button! You can use 3 different functions to handle the 3 buttons.
- 2.4. Save the file and make sure you keep the .html extension. If you are using a PC, remember to tell TextEdit not to put the ".txt" extension after ".html"!
- 2.5. Try opening the file with a browser and see how it works. If any of the elements above don't work, go back into the text editor window, and edit the file to fix them. The reload the file in your browser. Keep doing this until everything works just perfectly :-)
- 2.6. **Advanced Students:** Having three different functions in the header section seems a bit of a waste. Have your 3 buttons only use one function. But have that function behave differently

based on an “argument” or “parameter” that you pass to the function. An example of this technique can be found here:

- http://www.w3schools.com/js/tryit.asp?filename=tryjs_functionarg2

3. **Image maps and JavaScript**

3.1.2 Look at the following HTML file:

```
<html xmlns="http://www.w3.org/1999/xhtml">
  <head>
    <script type="text/javascript">
      function writeText(txt)
      {
        document.getElementById("desc").innerHTML=txt;
      }
    </script>
  </head>
  <body>
    
    <map name="colormap">
      <area shape="rect" coords="0,0,100,100" onMouseOver="writeText('Blue')"
        href="blue.html" target="_blank" alt="Blue" />
      <area shape="rect" coords="0,0,200,100" onMouseOver="writeText('Purple')"
        href="purple.html" target="_blank" alt="Purple" />
      <area shape="rect" coords="0,0,300,100" onMouseOver="writeText('Grey')"
        href="grey.html" target="_blank" alt="Grey" />
    </map>

    <p id="desc"></p>
  </body>
</html>
```

3.2. Use the example above to create a new document called jsimagemap.html. Modify the example above so that it uses the image file: [mymap2.jpg](#)

3.3. Replace the href=... sections of each <area...> tag with the same onclick="show_alert()" function(s) you used in section 2. You can use an argument to the show_alert() function if you know how, otherwise you will need 3 functions and each should have appropriate text.

3.4. Note that you will need to update the size of the image in the <img...> tag and the corresponding coordinates in each <area...> tag. Hint: The image is 200x200 pixels.

3.5. Also note that you will need to create a fourth <area... > tag since there are four colors in the image.

4. **Ready for more?**

4.1. Try working through the examples in the JavaScript notes posted on the class web page.

4.2. Type in the examples and test them to make sure they work. Then play with them, modifying them to do different things you can think of.

4.3. Your project for this section of the class will be an interactive animation. Can you modify the animation of the stick figure walking in place ([SimpleAnimation2](#)) on the website so that the figure will start/stop walking when a button is pushed?