**blackCSIS210 - Data Structures**

Web App. Dev.

### Laboratory 3

**Lab 1**

# Names \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

#

# General Lab Procedures

* You should create a directory (folder) in your home account called csis390. At the beginning of each lab, create a new sub-directory called labX, where X is the lab number.
* Files used in the lab can be found on the course canvas webpages.
* Turn in this lab sheet stapled to print outs of the code you produce as needed in each assigned section from the laboratory manual. These sheets should be in order. One lab submission is sufficient for each group.
* You can find documentation for the HTML 5 specification, the CSS, and the Javascript references respectively at

<https://html.spec.whatwg.org/>

<https://developer.mozilla.org/en-US/docs/Web/CSS/Reference>

<https://developer.mozilla.org/en-US/docs/Web/JavaScript>

**Lab Objectives**

* Practice with advanced css styling
* Use winSCP to move files from a local machine to a server.

## Lab 1

#### Part 1 – Button css

In this part of the lab, you will create an HTML webpage that includes a button with different formats. You will practice stylizing this button using css.

* Create an HTML page with Buttons as its title.

* According to the official HTML5 specifications, what are the three types of buttons?
* Within the body, create a button tag, so that a button appears in the middle of your page.
	+ Have the button have the text My First Button.
	+ How do you center your button using CSS? (Hint: Use a div)
	+ To accommodate people with poor eyesight, change the button’s font size to 24pt. Make the button itself bigger to accommodate this new font size.
	+ My favorite font is comic sans. Change the font type to this. If it is not available, use Tahoma instead.
	+ The rectangular edges to the button are too severe. Change it so that the corners are rounded.
	+ The greyish background is not original. Change the background color and text color to different values.
		- Colors can be represented by a pound sign, followed by three (instead of six) hex digits, such as #ABA. What is the six hex-digit equivalent of #ABA?
* Once completed, demonstrate your webpage for your instructor and have him initial here. If you do not finish during the lab period, then demonstrate your webpage at the beginning of the next lab period.

Instructor’s Initials \_\_\_\_\_\_\_\_

* Add hover effects to your button so that the colors of the text and the background color change to something else when the mouse goes over the button.
* The transition is immediate when you hover in and out of the button. Change the time it takes to transition from one color scheme to another.
* For kicks, add the functionality of changing the cursor look, when it hovers, to a wait (spinning circle in chrome) icon.
* Finally, add a dark shadow to your button to give it a 3-D look.
* Once completed, demonstrate your webpage for your instructor and have him initial here. If you do not finish during the lab period, then demonstrate your webpage at the beginning of the next lab period.

Instructor’s Initials \_\_\_\_\_\_\_\_

#### Part 2 – HTML and the DOM

In this part of the lab, you will create a graphical representation of a DOM for a sample webpage.

* Download from the Canvas main course page resources the file **Simple.html**.
* Examine the diagram picture example found at <https://www.w3schools.com/js/js_htmldom.asp>.
* Create a DOM Tree of elements of Simple.html, and **attach it to your lab submission**. You may draw it by hand, or use an electronic drawing program of your choice.
* If you were to write Javascript to add an additional link to the document following the link in the page, how would you do it?
	+ What is the javascript method that would create a new element?
	+ What is the javascript method that would add or append a new child element to the document?
	+ What are the javascript statements needed to add a link to the site <http://www.siena.edu> with the clickable text Siena College after the current link?

#### Part 3 – Form Styling & Validation

 In this part of the lab, you will be working with Javascript in order to validate form input.

* Download from Canvas the file **append.html**.
* Open append.html in Chrome. Note that it has a label and input text field.
* Open append.html in Notepad++ and notice how the label and input text field are created using JavaScript.
* Add JavaScript code to create a div, label and input field for entering Birthday and then append it into the form. Do not add HTML, instead add JavaScript code to generate the elements.
* Use type="date" for the input element.
* Use "pony\_birthday" as the name and id of the input element.
* Your form should look like this:

* Add embedded CSS so your form looks like this:

	+ Hints: Give the div elements a 1px solid gray border, add border-radius, padding and margin attributes with values. Make the label elements display block. Give the input elements a height of 1.6rem and a font-size of 1.1rem.
* We will now wrok with event handling. Save append.html as **event.html**
* Be sure to open event.html in your web browser. A common mistake is to edit one file, but test a different file in the web browser.
* Add a submit button using JavaScript. Note that a submit button is just an input element where type="submit"
	+ Use value="Insert" for the submit button
	+ Use "action\_button" as the name and id for the submit button.
* Your form should look like this:

* Add the following event listener code to the very bottom of your JavaScript code (but still inside the script tag):

ponyForm.addEventListener("submit", checkForm);

* Add the following function definition at the very top of your JavaScript code (but still inside the script tag):

 function checkForm(event) {

 alert("Submitting form");

}

* Save event.html and re-open/refresh it in Chrome.
* When you click the submit button, an alert window should pop up and if you click "OK" then the form will be submitted.
* Add more embedded CSS so your submit button looks like this:

	+ Hints: To select only the submit button used the advanced selector input[type~="submit"] and give it a height of 2.5rem and a lightgreen background-color. Add padding, border, border-radius, and box-shadow to approximate the image above.
* Add the following JavaScript code at the very bottom to create a hover effect on the submit button:


While we could have used CSS to achieve the same hover effect, this code illustrates how the CSS is implemented by web browsers.

* **Important:** In the code above this refers to the submit button. In general, this can be used to refer to the object that is the "subject" of the event. Rather than define separate functions for changing the submit button, we use an **anonymous function** because the function won't be reused.
* Verify that event.html is valid using an html validator.
* Once completed, demonstrate your webpage for your instructor and have him initial here. If you do not finish during the lab period, then demonstrate your webpage at the beginning of the next lab period.

Instructor’s Initials \_\_\_\_\_\_\_\_

* Finally, we will validate user input for this form. Save event.html as **valid.html.**
* Be sure to open valid.html in you web browser. I want you to get into the habit of making sure you are editing the same document that you are testing. This is one of the most common problems when encountering bugs you cannot fix. It is because you are not editing the file with the bugs.
* Add code to the checkForm function to make sure the Name is not blank, and to make sure the birthday year is valid. For testing purposes, a valid birthday year should be between 2000 and 2015.
* Your finished form should work like the one at <http://breimer.sienacs.com/courses/csis-390-s17/labs/lab3/valid.html>
	+ Note that I have minified (compressed) the JavaScript to obfuscate the solution. You cannot just copy and paste this JavaScript. You must write it yourself with meaningful variable names and proper indenting.
	+ Chapters 8 and 9 in the zyBook are JavaScript tutorials/references. You can use it to learn about the syntax of the language.
	+ All your code goes inside of the checkForm function.
	+ You can select elements using a CSS-like query selector:

var ponyNameElement = document.querySelector("#pony\_name");

* + You can get the submitted value and save it to a variable:

var submittedName = ponyNameElement.value;

* + You can prevent the form from being submitted by calling this method:

event.preventDefault();

* + You can modify the CSS of a selected element:

ponyNameElement.style.border = '5px solid red';

* + You can modify the CSS of a selected element's parent (Note that this would modify the pony name’s div:

ponyNameElement.parentNode.style.color = 'red';

* + You can select the label by using parent-child relationships and change the innerHTML to be the error message. Note that this would modify the first label tag inside the pony name’s div.

ponyNameElement.parentNode.firstChild.innerHTML = "Name is required"

* + You can use an if-else statement to switch the style back if the value is valid:

if (submittedName == "") {

 ponyNameElement.style.border = '5px solid red';

 }

 else {

 ponyNameElement.style.border = "1px solid gray";

 }

* + You can get the value of the submitted birthday, which is a String:

var birthdayString = ponyBirthdayElement.value;

* + This string is of the form 2017-1-30. You can split it into an array using the "-" as a delimiter

var birthdayArray = birthdayString.split("-");

* + You can compare the year as follows

if (birthdayArray[0] < 2000)

* Verify your HTML.
* **Create a zip file of your lab3 folder called lab3.zip and submit the file in Canvas by the beginning of the next lab period. The zip file should contain three files: append.html, event.html and valid.html. In the comment area of Canvas put your partner's name.**