Part 1

1. Syntax and semi-colons: (textbook page 74) semi-colons are optional but makes the code easier to read and allows you to have several statements on one line. We use the semi-colon to terminate a statement (Freeman, 2011).

<script = “text/javascript />

Document.writeln(“Hello World”);

</script>

2. Case-sensitivity and variable names (reserved names): Javascript is Case-Sensitive (Freeman, 2011). Use camo case.

var fieldsFilled = true;

3. Declaring variables and assigning values: (textbook page 77) You use the **var** keyword to define variables. You do not have to declare the type of variable. You can assign it to different types and you can assign a value it in a single statement. Local variables are defined in a function and are only available within that function. Global variables are defined in the script and can be accessed anywhere (Freeman, 2011).

<script type=”text/javascript”>

var global = “Shelby”

function myFunc(name) {

var local = “winter”;

</script>

4. Arithmetic, comparison and logical operators:

Arithmetic operators are + addition, - subtraction, \* multiplication, / division, % remainder (Freeman, 2011),

Comparison operators are ++, -- to increment and decrement, < less than, > greater than, <= less than or equal to, >= greater than or equal to, == equality, != inequality, === identity, !== nonidentity statement (Freeman, 2011).

The Logical operators are “and”, “or”, and “not” && for AND, || for OR, ! for NOT (Freeman, 2011).

5. Data Types: Number, String, and Boolean (University, 2012).

6. Concatenating Values: use + to concatenate a string (University, 2012).

7. Commenting: You can use // for a single line comment or /\* insert comment here \*/ for a multi-line comment (University, 2012).

8. Use of the Script tags and Linked JS files: the <script> tag is used to tell the computer that the program is separate from HTML and is allowed in the head and body of the document.

To link a JS file: <script src="myscripts.js"></script>

9. Creating a function and calling a function from an HTML element: to create the function you have to write its code, to the call the function you have to instruct the browser to execute the code controlled in the function. Examples of functions are Number, parseInt, and ParseFloat (Freeman, 2011).

10. Using Void: You can used when you want to stop the form (University, 2012).

11. Event Handlers for Buttons: intercepts the click and then processes a new file. Links such as onClick, onChanged, onSubmit: gives choices to click on, change, and submit. onClick is used in button, checkbox, radio, link, reset, submit, area. onChanged is used in select, text, textarea, and onSubmit is used in the form (Freeman, 2011).

12. Document Object Model: document tree and relation between the elements. Knowing siblings and descendants (Freeman, 2011).

13. Methods and Properties, InnerHTML: A method is a function that belongs to an object. You can use the **in** operator to check if an object has a property. You can add new methods to an object by setting the value of a property to be a function. You the delete key to delete a property or method (Freeman, 2011).

14. Navigator.appName: you can use to get the name of the browser, Navigator.appCodeName: use to get the code name of the browser, and navigator.userAgent: use it to get the user-agent header sent by your browser.

15. Window.Alert, Prompt: The window object is the very top level of the browser object hierarchy. Alert() and prompt() methods belong to the window object. It is used to open a new webpage, pop-up window, or gibing a message to the users. Prompt () is used to ask the user to enter information and Alert () is used to display the values of the document.

16. Document.write(): a built in method used for outputting content to the web page and used to display a message.

17. Location.href(): use it to change the Web page to the home page of another website.

18. Document.getElementByID(): accesses the first element with a specified ID.

19. Submit(): lets you add a submit button to submit a form

document.getElementById("form").submit();

20. Conditional Expressions: allow you to add a condition to return one expression depending on the conditions (Freeman, 2011).

21. Switch: can be used to perform different actions based on different conditions such as just one of many blocks of code (Freeman, 2011).

22. Loop (For, While): Loops are used to repeat steps,the **For** loop uses a counter variable, initialization, and termination, it is made to execute a fixed number of times. The **While** Loop is used to execute an unknown number of times and the programmer provides a way to tell the loop when to stop (University, 2012).

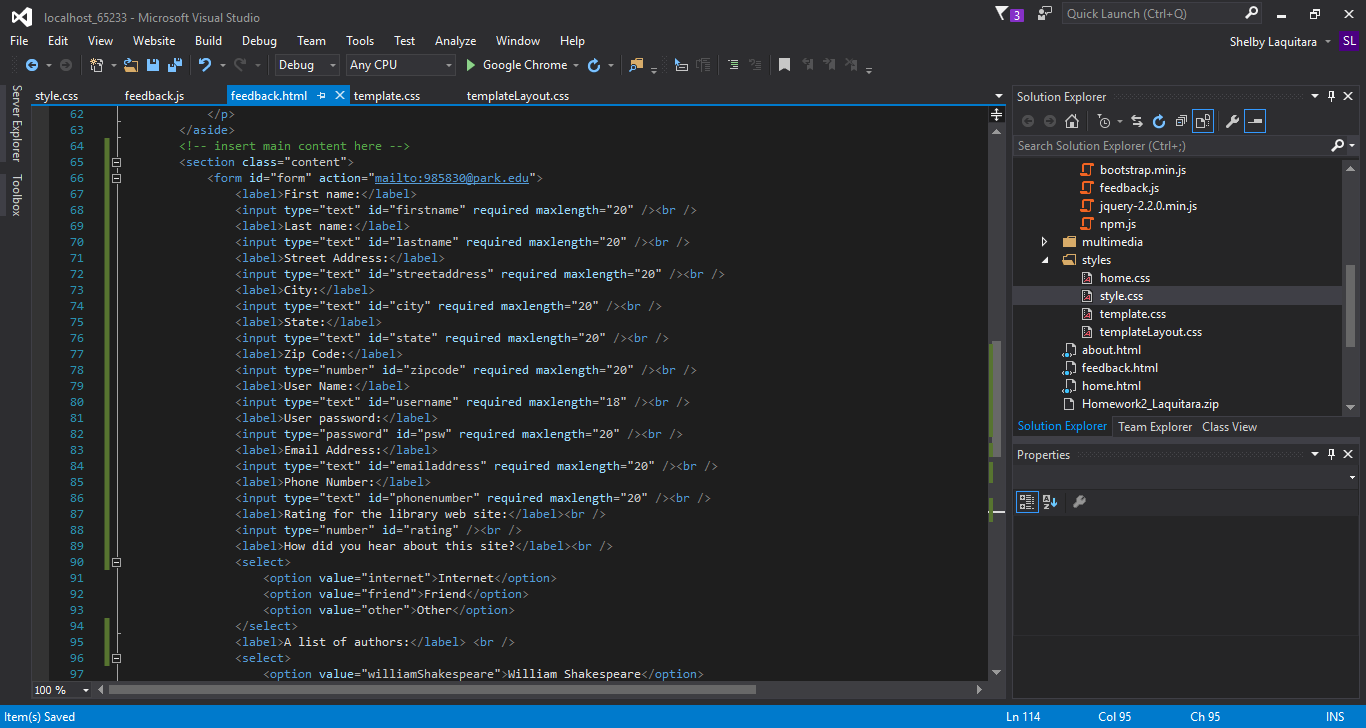
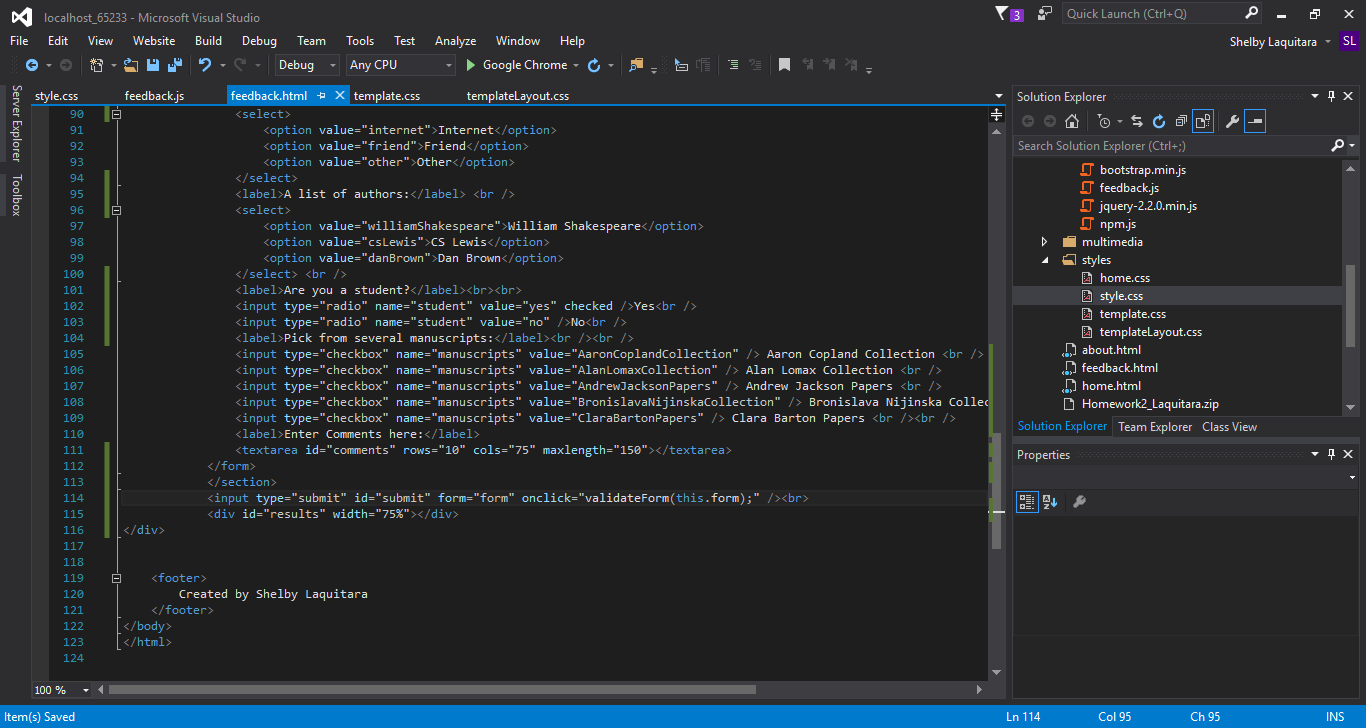
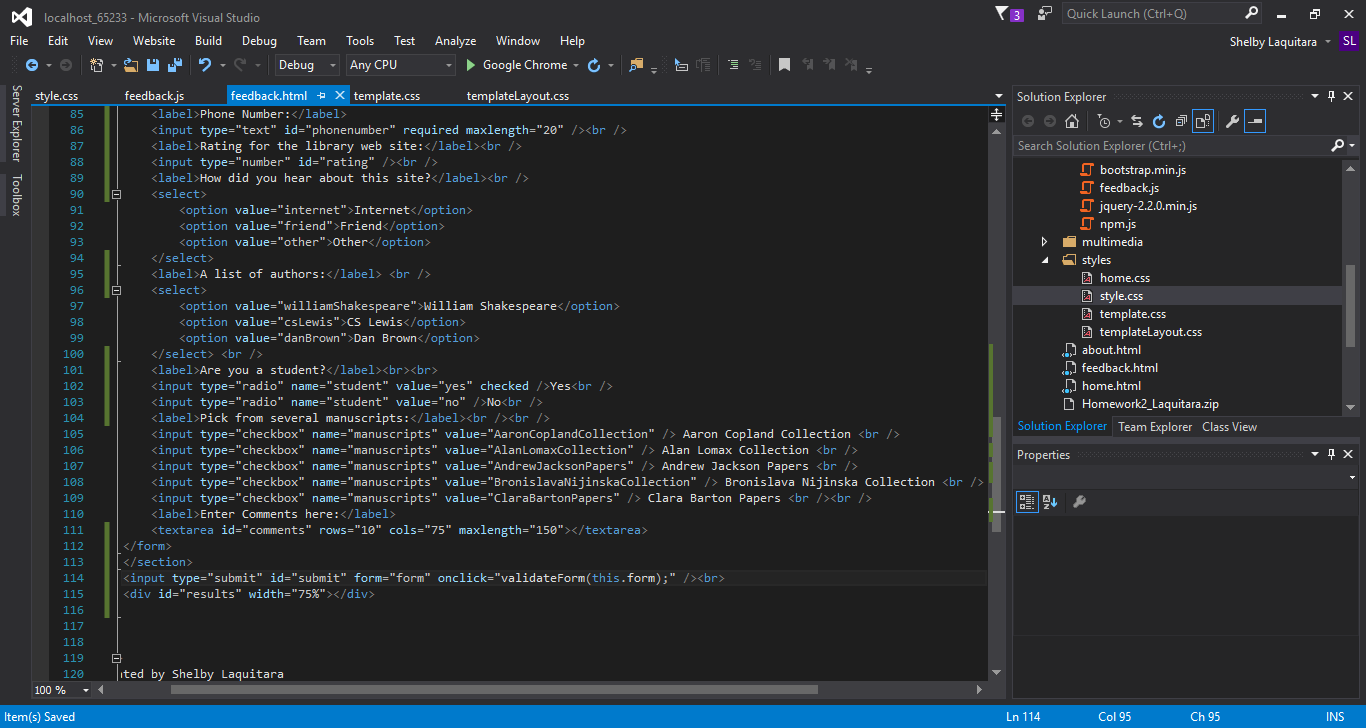
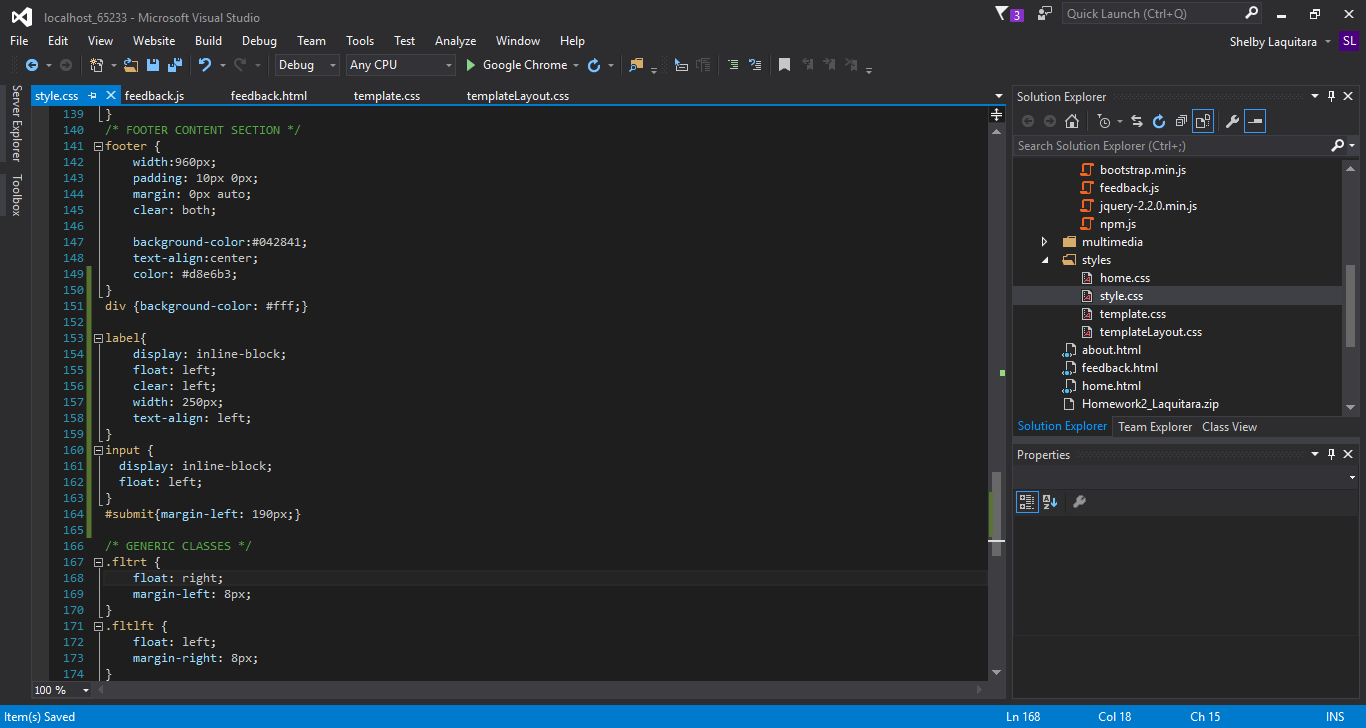
23. If Then: is a statement that works by testing a condition for true or false; if true then one block of code is executed, if false then the second block is executed instead (University, 2012).

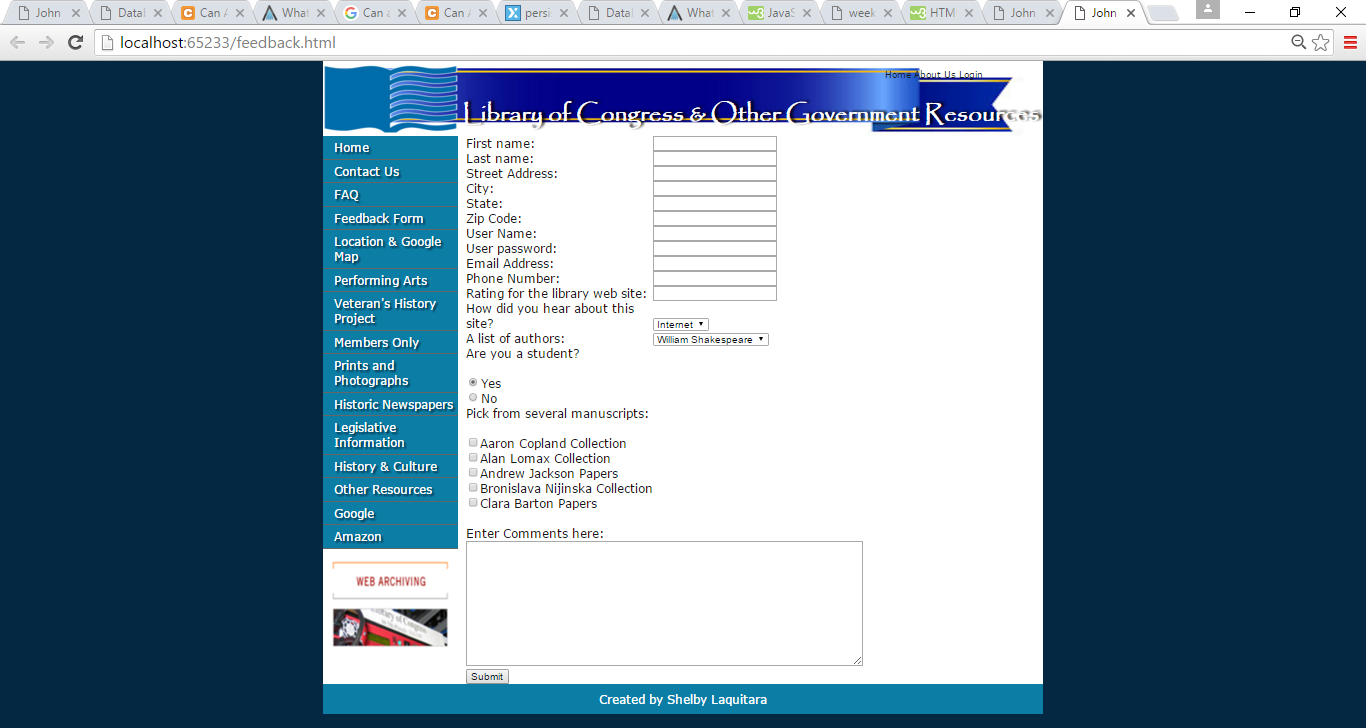
References

Freeman, A. (2011). *The Definitive Guide to HTML5.* New York: apress.

University, P. (2012). *Week 5 Lecture 5 Intro to Javascript.* Parkville.

Part 2-





Part 3-

