Announcements

- Midterm #2 scores are ready.
Validating through Web Developer

- Web Developer is an add-on (extension) for firefox
  
  https://addons.mozilla.org/firefox/60/

- Let’s see some of its options, including validation
Events

- **Event** – Notification that something has occurred
- Example of situations that make the web browser generate an event
  - Browser finishes loading a document
  - When the user clicks on a button
  - When the user moves the mouse
  - Others
- **Event handler** (also known as event listener)
  - JavaScript function or code fragment that is executed when a particular event occurs
- **Event handler registration**
  - Associating an event handler with a particular event
- **Example**: EventEx.html
Event-driven Programming

- Normal (control flow-based) programming
  - Approach
    - Start at main()
    - Continue until end of program or exit()

- Event-driven programming
  - Start at main()
  - Register event handlers
  - Await events & perform associated computation

- GUIs (Graphical User Interfaces)
  - Example of event-driven software
Event Handler Attributes for most HTML

- **Mouse Related**
  - `onclick` – mouse button is pressed and released
  - `ondblclick` – mouse button is double-click over element
  - `onmouseover` – mouse moves over element
  - `onmouseout` – mouse moves off element
  - `onmousemove` – mouse pointer is moved
  - `onmousedown` – mouse is pressed down while cursor is over the element
  - `onmouseup` – mouse is released while the cursor is over the element

- **Keyboard Related**
  - `onkeypress` – key pressed and released
  - `onkeydown` – key is pressed
  - `onkeyup` – key is released

- **Other**
  - Keep in mind that there additional handlers that are specific to certain tags. We will address those later on
For Loop

- Iteration statement
- General form

\[ \text{for (initialize; test; expression) } \]
\[ \text{statement} \]

basically equivalent to

\[ \text{Initialize} \]
\[ \text{while (test) } \]
\[ \text{statement} \]

If more than one statement use \{ \} 

- **Example:** ForLoops.html
- **Example:** ForLoopVariations.html
Problem - You need to keep track of the scores of students in a class
- Declaring and handling 50 variables is not an easy task
- Arrays come to the rescue

Array – Collection of values that can be treated as a unit or individually.
- You can visualize an array as a set of variables one after another
- There are several ways to define arrays.

```javascript
var scienceScores = new Array();  // Creates an empty array
var mathScores = new Array(3);    // Creates an array with 3 entries
var englishScores = [77, 88, 65]; // Creates an array with 3 entries
                               // having the specified values
```
Arrays

- To access elements of an array
  - Use the [ ] operator
  - We will use index values **starting at zero** to represent each element
- Accessing array elements

```javascript
mathScores[0] = 70; // Assigning 70 to the first array element
mathScores[1] = 80; // Assigning 80 to the second array element
var total = mathScores[0] + mathScores[1]; // reading the first
  // and second elements
```

- The array length property defines the number of elements
- Several functions are associated with arrays.
  - sort() – sorts elements of an array
  - reverse() – reverse elements of an array
  - join() – converts elements of an array to string and concatenates them
  - Others
- For loops are frequently use to iterate through arrays
- **Example:** ArrayEx.html