Announcements

- Instructor: Nelson Padua-Perez (nelson@cs.umd.edu)
- No posting of code in the forum
- Check class announcements daily
- You must implement programming projects by yourself
Link/CSS Example

- **Example:** Links folder
Conventions to Use From Now on

- Variable/Function names
  - We will use lowercase
  - If multiple words are associated with a variable name then capitalize the first letter of second word on
    - `waterTemperature`
    - `globalWarmingIndex`

- Curly Brackets
  - Use a particular style

- Comparisons
  - Use `===` rather than `==`

- No global variables!

- Good indentation
Introduction to Functions

- Function ➔ An entity that completes a particular task for us
- It can take values necessary to complete a particular task
- After completing a task it returns to the point after the call
- Examples of JavaScript functions
  - `document.writeln`
  - `alert()`
- You can define your own functions.
- Order of declaration is immaterial
- **Example:** Functions.html
Introduction to Functions

- General form of a function is:

```javascript
function name (<comma-separated list of parameters>) {
  statements
}
```

- Functions are invoked (called) by using the () operator
- A function can receive values via parameters
  - We don’t use var for parameters
- A function may return a value
- There are other approaches to define functions
Scope of Variables

- Variables declared in a function are called local variables.
- They are created on entry to the function and destroyed on exit.
- You can use the same name in different functions as they are different variables.
- Variables declared outside of a function are called global variables.
Functions Returning Values

- A function can return a value via the return statement
  
  \[ \text{return expression;} \]

- A call to a function that returns a value can be used as an expression

- The function execution terminates when a return statement is executed

- A return statement with no return value terminates the function execution

- Can we return more than one value?

**Example:** FunctionReturn.html

- Can we reduce the code for the maxValue function?

- What if we want to compute the maximum of more than two values?
JavaScript (Functions)

- Advantages of functions are:
  - Allows you to factor out common code
  - Allows you to reuse code
  - Allows you to control the code complexity
- While designing a solution to a problem you can divide a problem into sub-problems each represented by a function
main() Function

- The organization for code dealing with functions will be as specified in the following example

  **Example**: MainFunction.html
Global Variables

- Global Variables → variables defined outside of any function
- We want to avoid using global variables. Why?
Testing/Debugging

Testing
- Remember to test your code as you develop it
- First use simple data sets

Debugging
- Make sure your input is correct
- Use Error Console to track down problems
- Use lint to check the syntax
- Use alert to display values and identify the execution path
- Use trace tables
  - When writing trace tables create a column for each variable
    - Define the columns as you see fit, find the variables
- When writing functions
  - Test each function individually with its own driver
  - Let’s see an example
<br/> vs. 

- What is the difference between the two?
- When should we use them?
- Example: Brvsnewline.html
document.writeln/write

❖ Example: WriteLnVsWrite.html
Passing Values to Function

- Mechanism used to pass values to function is called *pass-by-value*
- Parameters
  - Variables that receive data
  - There are normal variables
- Arguments
  - Values you pass to a function
- Example: PassByValue.html
- Does it matter how we name the parameters?