# Assignment 6: Input and Output using JavaScript

## Objectives

* Create websites which take in input via **prompt** and put that information back out to the screen in various ways.

## Instructions

### Part 0: The Setup.

Create a folder on your hard disk, name the folder lastname\_firstname\_assignmentNo. You must replace “No” with the assignment number. Save all the files from this assignment in this folder.

There are several parts to this assignment. Create the following subfolders (in the folder lastname\_firstname\_assignmentNo (No is your assignment number)): q1, q2. As a result, you should have the following folder (directory) structure for this assignment: (-2 points if wrong)

* lastname\_firstname\_assignmentNo\q1\
* lastname\_firstname\_assignmentNo\q2\

Each question in this assignment require you to write a website that behaves as an application. The style for each page will be the same, but the HTML and the JavaScript will be different. The focus of this assignment isn’t CSS. In fact, you can use the same CSS code that you wrote from a previous assignment. It is important that each of your pages have a style to them. Bland websites won’t work. Also important: no two students should have the same CSS code.

Each page should have the following components:

* A header section in which you announce the name of your application.
* A main which contains your application.
* A footer in which you present your name and the copyright year. This should be the same for every question.

These specifications are detailed enough for you to complete each questions. However, you’ll need to use your best judgment on how to code many of the details. To do that, write the code in the way that you think is best, based on the skills that were presented in the materials. Any time I refer to the title of a page, that should include the title bar and the content of the header section.

### Question 1. 50 points.

Create a website and store it in the “q1” folder. The file should be named “index.html”. I want you to pick out a favorite book.

* Make the title of this page: “(Your Name)’s Book App”. This should go in the title bar and the header section of your code.
* Include the necessary html documentation using comments. Make sure that you include your name, class number, section number, and the date.
* Create three empty “div” elements on the screen. Give the following id’s to each element: title, author, and subject. Make sure that each “div” element is properly labeled as a book title, an author, and a subject.
* Put a single button on the screen. When the user clicks the button, **prompt** the user:
  + **prompt** for the title. Store this in a variable.
  + **prompt** for the author. Store this in a variable.
  + **prompt** and the subject matter (fiction, science, self-help, religion, etc.). Store this in a variable.
  + (For example, I might respond with “Alice’s Adventures in Wonderland”, “Lewis Carroll”, and “Fiction”.)
  + Write the book title to the title element, the author to the author element, and the subject matter to the subject element. The three areas should be clear to anyone reading this website and there should be no ambiguity as to which area is the title, author, and subject of the book.

What should you name your variables? That is for you to decide, but it should be clear to whoever is reading your code what is contained inside the variable by the name.

### Question 2. 50 points.

Create a website and store it in the “q2” folder. The file should be named “index.html”. I want you to write a program that will play with people’s ages.

* Make the title of this page: “(Your Name)’s Age App”. This should go in the title bar and the header section of your code.
* Include the necessary html documentation using comments. Make sure that you include your name, class number, section number, and the date.
* Create a single text input element on the screen that asks the user to enter their age.
* Put a submit button on the screen.
* Put two div elements on the screen. The first will be labeled “addFive” (for the user’s age in five years) and the second will be labeled “timesTwo” (for the user’s age doubled.)
* Put a submit button the screen. When the user clicks this button:
  + The program pulls the data out of the input field. It is assumed that this will be numerical data.
  + The program will convert this value to a number type.
  + The program will display to the screen the user’s age in five years in the “addFive” div element.
  + The program will display to the screen the user’s age doubled in the “timesTwo” div element.
* For example. Assume that I am 100 years old. In five years, I will be 105. My age doubled is 200.
* For example. Assume that I am 25 years old. In five years, I will be 30. My age doubled is 50.
* If your program works with any positive integer and follows this pattern, you have completed the assignment.

## Turn it in

* Create a PDF document of screenshots of each of your html pages. Also, include a screenshot of all pages after being passed through the HTML validation service. This should be (2 screenshots times N questions) screenshots total.
* I will deduct 20 points for every error and 2 points for every warning when validating your files.
* **Important**: Part of web design is that your websites must be readable. If anything on your page is not 100% readable, I will be deducting points. For example, if there is text on a background and the background makes the text unreadable, I will deduct points. If text flow is obstructed by an image, I will deduct points. If anything about your page makes it difficult to read the content, I will deduct points. If something about the interface (either input prompts or output) is not obvious to someone running your code, I will deduct points.
* There are several critical parts of this assignment that (if missing) will automatically fail the assignment:
  + The page titles don’t contain your name.
  + You fail to put your name in the document comments for the first HTML file.
  + You fail to include the PDF document of your screenshots.
* You need to upload exactly 2 files in the same upload:
  + **Zip up your folder** containing your HTML, CSS, and image files (if applicable) into a single ZIP file and upload it.
  + Upload your PDF file independent of your ZIP file.