ITSE 2302 – Program 03

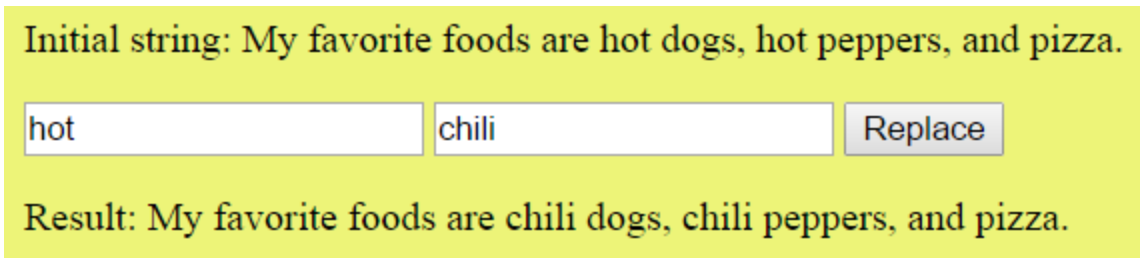
All of the assignments in this course must be uploaded to Bb **and** *published to your student website*. Use the numbered list below **and** the Common Requirements as a confirmation checklist.

- Complete the steps described at the Website Setup & Publishing link in the "Assignments" section in Bb.
- Use the course material located at
  - HTML Tutorial , CSS Tutorial , JavaScript Tutorial
- Program 03 can be completed using content from these tutorial sections:
  - HTML HOME – DOM Node Lists
- Develop the page/site for your user – Communicate with your user.

Requirements (remember to identify the requirements by #*number*):

Note: the requirements below refer to the index.html within the program-## folder unless otherwise stated.

1. The Program 03 link on your top-level page should open an index.html page in the program-03 folder.
   a. user enters 2 strings
   b. use a regular expression to search for the second string within the first string (see this link for code example).
   c. add the "use strict" directive at the top of the program-03.js file to prevent hoisting (see JS Hoisting)
2. Start with an initial string (for example): "My favorite foods are hot dogs, hot peppers, and pizza." User enters 2 strings. Use a regular expression to replace all occurrences of the first string with that of the second string within the initial string. Hint: see the modifiers. Example:



3. User enters a number. Check if number is between 200-7000 inclusive. Use try-catch-throw to produce an error message if the entry is not a number. Also, throw an error if it is too high or low and display to the user how far above or below the range their number was. Include a finally statement.
4. Include an example to demonstrate the console.log() method and the debugger keyword.
5. Include a numbered list listing the primary components (approx. 12) of the JavaScript Style Guide. Make each list item selectable. When clicked, a window pops up displaying 1-2 sentences about that style guide convention.
6. Optional: Use the HTML Application Cache API to make a page with a script and an image available offline. Name the page "test-appcache.html". Include a link to the page and instructions for the user to demonstrate the cached behavior. Note: Since the HTML Application Cache API has been deprecated, this requirement (Requirement 6) is no longer a "Requirement" and can be considered optional. It is only included here to ensure exposure to the topic.
    a. Even though it is part of the HTML5 specification, it has been deprecated and is not recommended for use.
    b. You do NOT need to study the topic for exam purposes since it has been removed from the W3Schools content.
    c. MDN
    d. CanIUse
7. Include a numbered list listing the primary components (approx. 11) of the JavaScript Best Practices. Make each list item selectable. When clicked, a window pops up displaying 1-2 sentences about that best practice.
8. Convert a JSON object into a JavaScript string for use in JavaScript:
    a. JSON object contains the restaurant array which contains 4 objects (restaurants) with name:value pairs.

   b. Each object in the array consists of the following name:value pairs: restaurantName:value, cuisineType:value, qualityRating:value (value of 1-10 that you include).
   c. Use JSON.parse() to convert the string into a JavaScript object.
   d. Use iteration to display the contents of the JavaScript object.
9. Include a form:
   a. that has firstname and lastname textboxes
   b. calls a function to validate the form onsubmit
   c. sends an alert if either name textbox is empty
   d. Use action=""
10. Include a form like that above but uses only HTML to validate.
11. Include a function that uses the following validity properties (include supporting HTML element):
   a. rangeOverflow
   b. rangeUnderflow
   c. valueMissing
12. Create 3 objects. One by each of the techniques listed:
   a. object literal
   b. keyword new
   c. object constructor

   Add the objects to an array and use iteration to display the contents of the objects.
13. Add a new property to one of the objects previously created and display the property. After displaying, delete the new property. Naturally, be sure to message to the user that the property was deleted.
14. Create a new "aircraft" object (via object constructor) that includes (among others) a speed property. Include a changeSpeed() method that will increase or decrease the aircraft speed based on the argument sent in the function call. For instance, changeSpeed(10) will increase speed by 10 knots and changeSpeed(-10) decreases speed by 10 knots. User enters the number and clicks a button to call the method.

15. Use the prototype property to add a new property (range) and a new method (addFuel) to the aircraft prototype. As always, display results.

TEST – TEST – TEST your **website** to ensure the requirements are met.

1. Use the list above **and** the Common Requirements as a confirmation checklist.
2. Not meeting all requirements = 0 points for the assignment.
3. Complete the appropriate Program ## confirmation in Bb by the due date.
4. Upload your .zip file to Bb under the appropriate Program ## link by the due date.
5. Ensure your assignment functions correctly on your student website and that all requirements are identified by "#XX".
6. Remember to study for the exam. Distributing learning over time is far superior than panic-study in close proximity to the exam date.