

**CSE2201 Lab 6**  
**Department of Computer Science**  
**Faculty of Natural Sciences**  
**University of Guyana**

**Semester II, 2017-2018**

---

**Description:**

In this assessment you'll be tasked with taking some of the knowledge you've picked on javascript and applying it to creating a fun app that generates random silly stories. Have fun!

**Starting Point:**

To get this assessment started, you should go and grab the raw text you need to mark up, and the CSS you need to include in your HTML. These assets are available at: <https://bitbucket.org/ChrisIsKing/cse2201/src> in the lab 6 folder.

**Lab Brief:**

You have been provided with some raw HTML/CSS and a few text strings and JavaScript functions; you need to write the necessary JavaScript to turn this into a working program, which does the following:

- Generates a silly story when the "Generate random story" button is pressed.
- Replaces the default name "Bob" in the story with a custom name, only if a custom name is entered into the "Enter custom name" text field before the generate button is pressed.
- Converts the default US weight and temperature quantities and units in the story into UK equivalents if the UK radio button is checked before the generate button is pressed.
- Will generate another random silly story if you press the button again (and again...)

Steps:

**Basic setup:**

1. Create a new file called main.js, in the same directory as your index.html file.
2. Apply the external JavaScript file to your HTML by inserting a <script> element into your HTML referencing main.js. Put it just before the closing </body> tag.

**Initial variables and functions:**

1. In the raw text file, copy all of the code underneath the heading "1. COMPLETE VARIABLE AND FUNCTION DEFINITIONS" and paste it into the top of the main.js file. This gives you three variables that store references to the "Enter custom name" text field (customName), the "Generate random story" button (randomize), and the <p> element at the bottom of the HTML body that the story will be copied into (story), respectively. In addition you've got a function called randomValueFromArray() that takes an array, and returns one of the items stored inside the array at random.
2. Now look at the second section of the raw text file — "2. RAW TEXT STRINGS". This contains text strings that will act as input into our program. We'd like you to contain these inside variables inside main.js:
  - I. Store the first, big long, string of text inside a variable called storyText.
  - II. Store the first set of three strings inside an array called insertX.
  - III. Store the second set of three strings inside an array called insertY.
  - IV. Store the third set of three strings inside an array called insertZ.

**Placing the event handler and incomplete function:**

1. Now return to the raw text file.
2. Copy the code found underneath the heading "3. EVENT LISTENER AND PARTIAL FUNCTION DEFINITION" and paste it into the bottom of your main.js file. This:
  - Adds a click event listener to the randomize variable so that when the button it represents is clicked, the result() function is run.
  - Adds a partially-completed result() function definition to your code. For the remainder of the assessment, you'll be filling in lines inside this function to complete it and make it work properly.

**Completing the result() function:**

1. Create a new variable called `newStory`, and set its value to equal `storyText`. This is needed so we can create a new random story each time the button is pressed and the function is run. If we made changes directly to storyText, we'd only be able to generate a new story once.
2. Create three new variables called `xItem`, `yItem`, and `zItem`, and make them equal to the result of calling `randomValueFromArray()` on your three arrays (the result in each case will be a random item out of each array it is called on). For example you can call the function and get it to return one random string out of `insertX` by writing `randomValueFromArray(insertX)`.

- Next we want to replace the three placeholders in the `newStory` string — `:insertx:`, `:inserty:`, and `:insertz:` — with the strings stored in `xItem`, `yItem`, and `zItem`. There is a particular string method that will help you here — in each case, make the call to the method equal to `newStory`, so each time it is called, `newStory` is made equal to itself, but with substitutions made. So each time the button is pressed, these placeholders are each replaced with a random silly string. As a further hint, the method in question only replaces the first instance of the substring it finds, so you might need to make one of the calls twice.
- Inside the first `if` block, add another string replacement method call to replace the name 'Bob' found in the `newStory` string with the `name` variable. In this block we are saying "If a value has been entered into the `customName` text input, replace Bob in the story with that custom name."
- Inside the second `if` block, we are checking to see if the `uk` radio button has been selected. If so, we want to convert the weight and temperature values in the story from pounds and Fahrenheit into stones and centigrade. What you need to do is as follows:
  - Look up the formulae for converting pounds to stone, and Fahrenheit to centigrade.
  - Inside the line that defines the `weight` variable, replace 300 with a calculation that converts 300 pounds into stones. Concatenate `'stone'` onto the end of the result of the overall `Math.round()` call.
  - Inside the line that defines the `temperature` variable, replace 94 with a calculation that converts 94 Fahrenheit into centigrade. Concatenate `'centigrade'` onto the end of the result of the overall `Math.round()` call.
  - Just under the two variable definitions, add two more string replacement lines that replace '94 fahrenheit' with the contents of the `temperature` variable, and '300 pounds' with the contents of the `weight` variable.
- Finally, in the second-to-last line of the function, make the `textContent` property of the `story` variable (which references the paragraph) equal to `newStory`.

**Hints & Tips:**

- You don't need to edit the HTML in any way, except to apply the JavaScript to your HTML.
- If you are unsure whether the JavaScript is applied to your HTML properly, try removing everything else from the JavaScript file temporarily, adding in a simple bit of JavaScript that you know will create an obvious effect, then saving and refreshing. Eg. `Console.log()`;
- `Math.round()` is a built-in JavaScript method that simply rounds the result of a calculation to the nearest whole number.

**Example:**

The following screenshot shows an example of what the page might look like after being marked up.

Enter custom name:

Chris

US

☒

UK

☐

Generate random story

It was 94 fahrenheit outside, so Big Daddy went for a walk. When they got to the White House, they stared in horror for a few moments, then turned into a slug and crawled away. Chris saw the whole thing, but he was not surprised — Big Daddy weighs 300 pounds, and it was a hot day.