| Sprint 8, Assignment 8.8 Please also update the doc name with correct numbers. | |
| --- | --- |
| Assignment type: JS Interactive | |
| Assignment name  Can remain the same as the assignment topic name, or…  can be created depending on the task in the active form, e.g. “Create your own X” | "Template Literals" |
| BDG Description\*  What is the task and why is it important?  In this part, it’s encouraged to think about storytelling and future job-specific context e.g. “You’ve been asked to help out X with Y. They want Z on their website, yet aren’t too sure on how to achieve it”.  Drawing on practical examples and adding context can increase a student's motivation and increase long term learning according to Instructional Design principles, because this helps to relate some familiar or existing knowledge to new bits of information.  This will appear in the course as text before a button, leading to the interactive platform assignment. | The autoparts supplier has asked that you update the language that will be displayed on their welcome page: they would like something more personal and engaging for the customer. Now that you know how to use methods, primitive data types, and variables let’s code more dynamic language for the welcome page! |
| The Assignment  A short specific description of the assignment and tasks using bullet points that the student will need to do. | Use a template literal (and one or more substitutions) to perform interpolation of   * + A variable into a string,   + A mathematical equation into a string, and   + A method on your   Also, add “use strict” to your file and re-run your code to make sure it still works (and to see if you are developing any natural, but bad habits!) |

| Steps  Step-by-step instructions on what the student should do. | | |  |
| --- | --- | --- | --- |
| Step # | Step  Write each small step of the task | At least 1x hint(s)  Write some text (not necessarily, but can also be a part of code if relevant) which would hint the student to figure out the correct step forward. | The correct output should be…  (if relevant to the task) |
| 1 | Add a smaller string into the middle of a larger string using a template literal | Store the customer’s name in the variable **customerName**, and then reference it in your template literal  let customerName = "Mariko";  console.log(`Hello **${customerName}**! You know what we always say …`) | `Hello Mariko!  You know what we always say …` |
| 2 | Perform math within a string, using a template literal | Insert a sentence with a template literal that performs math to display how long the customer has been a member  let customerName = "Mariko";  console.log(`Hi ${customerName}!  **It’s been ${30-3} days since you joined our family** - and you know what we always say: "We have better prices than our competition - \*every\* time!"  Make sure to check our “latest discounts” page!`); | `Hi Mariko!  It’s been 27 days since you joined our family - and  you know what we always say: "We have better prices than our competition - \*every\* time!"  Make sure to check our “latest discounts” page!` |
| 3 | Use a method with our variable inside of the string (using a template literal) | Let’s grab the customer’s attention by yelling their name and personalizing our last line for them (using the .toUpperCase method)  let customerName = "Mariko";  console.log(`Hi ${customerName}!  It’s been ${30-3} days since you joined our family - and you know what we always say: "We have better prices than our competition - \*every\* time!"    **And ${customerName.toUpperCase()} - make sure to check our “latest discounts” page!**`); | `Hi Mariko!  It’s been 27 days since you joined our family - and you know what we always say: "We have better prices than our competition - \*every\* time!"    And MARIKO - make sure to check our “latest discounts” page!` |
| 4 | Add “use strict” to the beginning of your file and re-run |  |  |
| 5 |  |  |  |
| 6 |  |  |  |
| ... |  |  |  |