|  |  |
| --- | --- |
| Sprint 9, Assignment 1.8 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” | “Create an Object Using Object Initializers” |
| 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 job boards have presented a new, although odd opportunity: a gentleman claiming to be “St Nicholas” would like you to digitize his user database. Ol’ Santa needs a record for every child, and to store in those records a number of different pieces of information. He also needs to assign a case worker - an “elf”, as he’s apparently calling them, to each child. And he wants you to build a demo file ASAP. |
| The Assignment  A short specific description of the assignment and tasks using bullet points that the student will need to do. | * Create an object with properties, * Define those properties, and then * Change at least one property. |

|  |  |  |  |
| --- | --- | --- | --- |
| 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 | Declare an object with one property (Key:Value pair) | Start with the managing elf on this project – maybe named Buddy | let kid = {          elfManager : `Buddy`,      } |
| 2 | Log a) the object, and then b) the value | You want to make sure both that you’ve declared your object properly, and that you can access individual properties of that object, so console.logging object.property is important | console.log(kid.elfManager); |
| 3 | Add four other properties | You need at least one of each of the following values: string, array of strings, boolean, number, and variable. | let kid = {          elfManager : `Buddy`,          firstName : `Cindy`,          lastName : `Lou`,          nice : true,          nocppy : 16, *// NoCPPY = "Number of Cookies Provided Previous Year"*      } |
| 4 | Make two changes to your object:  a) elfmanager’s value to a variable, and  b) combine first and last names into a single key with an array of strings as its value | In order to use a variable in your object, declare a variable like "elf007" before your object, set it to your desired name, and then make that variable the value in your object. | let elf007 = `Buddy`;      let kid = {          elfManager : elf007,          fullName : [`Cindy`,`Lou`],          nice : true,          nocppy : 16, *// NoCPPY = "Number of Cookies Provided Previous Year"*      } |
| 5 | Log your object, and your array of strings to terminal |  | console.log(kid);      console.log(kid.fullName); |
| 6 |  |  |  |
| ... |  |  |  |