|  |  |
| --- | --- |
| Sprint 9, Assignment 7.5 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 a JS pattern” |
| 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. | Create a JS pattern by re-writing properties of the prototype object |
| The Assignment  A short specific description of the assignment and tasks using bullet points that the student will need to do. | * Using .prototype method, redefine or add the functionality of all your objects |

|  |  |  |  |
| --- | --- | --- | --- |
| 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 key and value to the prototype object | Hint: if changing a constructor's prototype attributes, invoke .prototype on that constructor function  Hint: First, call the constructor for which you'd like to change its prototype. Follow this with the 'prototype' method. Finally, provide the name of the key, and have it equal a value. | deliveryRecipiants.prototype.prototypeGreeting = `MeEerry Christmas!`;      console.log(deliveryRecipiants.prototype.prototypeGreeting) |
| 2 | Create a new instance from your constructor, and check to see if it's prototype has the new k:v pair! | Hint: you will need to un-hide the prototype chain in order to see your new key:value in the console | const recipiant2 = new deliveryRecipiants("elfSUPREME", [`Boris`, `Spyder`], false, -5, [], ["pete townshend", "john entwistle"])      console.log(recipiant2); |
| 3 | Bonus: can you add a function to the prototype object, and then call it (like in the previous video)? |  | const thirdGreeting = {          sayThirdGreeting: function() {              return "and a HAPPPY NEW YEAR!"          },      }      deliveryRecipiants.prototype.newYear = thirdGreeting; |
| 4 |  |  |  |
| 5 |  |  |  |
| 6 |  |  |  |
| ... |  |  |  |