Goal7 : Assignment: Objects (3h)

Ready for the assignment on Objects?

Objectives & Outcomes

Successful completion of this activity will show that you:

- understand the details of "Object Instantiation" & "Object Prototypes" (how it works, when to use it, understand the formatting)
- exhibit working knowledge of "Object Instantiation" & "Object Prototypes", by writing JavaScript code that displays this knowledge
- display the ability to take the "Object Instantiation" & "Object Prototypes" and implement it into a program

Level of Effort

This activity should take approximately 195m to complete. It will require:

- 0m Research
- 15m Prep & Delivery
- 180m Work

If you find that this activity takes you significantly less or more time than this estimate, please contact me for guidance.

Reading & Resources

Objects – Rubric *(necessary)*
This rubric outlines the points for the assignment. Make sure you check off each one as done before submitting your assignment.

Assignment Files *(necessary)*
This link has all the files you need to get started with this programming assignment.

Instructions

Getting Started:

- In your local branch of the PWA1 GIT repo, you will need to create a directory entitled "goal7_assign_objects".
- Download the .zip file from FSO for this assignment and unzip it in the directory above, "goal7_assign_objects". This is where all your assignment development files will reside.
- Using JavaScript comments, place your name, date and assignment at the top of the JavaScript file.
- Make sure you adhere to proper folder constructs, if applicable (css, images, js, etc)
  (Online) Please watch the associated screencast that shows the finished assignment and explains the rubric and requirements

Criteria:

To obtain full credit on the assignment, the following criteria must be adhered to as well as satisfying all items on this assignment's rubric.

- This assignment will require the creation of two js files, **main.js** and **person.js**
- The **main.js** file will contain the following:
  - Create an array called **names** that contains at least 5 people names that will be eventaully chosen for each person.
  - Create three instances of the Person object using a for loop and place each reference of the object in an array called people.
  - When instantiating the Person object, make sure a randomly chosen name (using the Math.random() method) from the names array is sent to the constructor of the person along with what row number in the HTML the information will be displayed in.
    - Ex: var person = new Person(names[someName],someRow);
  - Create a function called "populateHTML" which outputs the person's name and person's job, in the DOM.
  - The code must be built to NOT allow duplicate names to appear.
  - Set up an Interval that calls a runUpdate() function 30 times a second. Example: "setInterval(runUpdate, 1000 / 30);"
  - Once all the Person instances are in the people array then loop through each person and run the prototype update() function. This loop should be done in the runUpdate() function that is called from the setInterval (in the example directly above). For example:
    - function runUpdate(){

      people.forEach(function(element){

      element.update();

      });
- The **Person.js** file will contain the following items:
  - Add the Person object to the global window object – because main.js needs it. Example: "window.Person=Person;"
  - Two variables, "jobs" and "actions" which should be directly on the Person object (static variables). The jobs variable is an array of 4 or more jobs. The actions variable is also an array of some actions a person could do.
    - ex: Person.jobs = ["teacher","farmer","student","pilot"];
    - ex: Person.actions = ["sleeping","eating","working"];
  - The Person constructor should also have four properties:
    - **name:** The name of the person. This property is set in this constructor for the Person.
    - **action:** This property is set in this constructor for the Person. This property states what the person is actively doing and it is one of the values in the actions array. You will randomly select one item from the Person.action array for this property (use the Math.random() method).
    - **job:** This property is set in this constructor for the Person and is one of the values in the jobs array. You will randomly select one item from the Person.jobs array for this property(use the Math.random() method).
    - **row:** Like the name property, this is set in this constructor.
    - Main.js writes out the information for the person's name and job. You will now need to display the initial action of the person in 3rd column.
  - The Person object must have an update() function added to its prototype. The purpose of this update() function, which is called from the main.js file, is to change the actions of the person every so often, this is based on the interval instructions given above (approx. once every 5 seconds or so) and display that change in the HTML in column 3.

Deliverables

You will submit your project via Git and github.com. This is where your work will be graded from. As part of your grade you are required to make at least 6 meaningful commits.