Adena Glasper

Week 3 Research

June 17, 2022

Array Methods

Arrays allow for programmers to be able to group similar data together without having to create a new variable over and over again. With an array you can refer to the value via its location and be able to access that index each time by creating very few lines of code. But, at times, arrays need to be able to be altered or manipulated and we have to be able to use actions in order to make these changes without unnecessarily redoing the array over. In order to alter an already created array, we would use array methods. There are 5 essential methods we can use, which are: map, reduce, for each, filter, and splice.

The map method allows “you to invoke a function for each element in an array and create a new array from the returned values of each iteration” (Promineo Tech 01:00–03:30). The map methods signature in order to enable it is taking the array name and adding a dot with map at the end right before the function starts. It would look like this: let homeworkDays = [“Monday”, “Wednesday”, “Friday”] followed by let lengths = homeworkDays.map. By using the map method, we are able to create a new array based off its original. The reduce method is useful for condensing all the elements of an array down to a single value. You will only want to use it when “you need to perform an operation on all elements in an array that results in a single value” (Promineo Tech 03:40–05:15). The reduce method’s signature would look like similar in that it shows let homeworkDays = lengths.reduce and then the function following after. Similar to the map method, you could use the for each method to invoke a function, but it does not return a new array. Using the for each method is good when we want to still perform the action of the function of each element in the array but have no use for storing the resulting value. The for each method’s signature follows as homeworkDays.forEach followed by function.

Filter method is another array feature that can be used to invoke a function but as a Boolean value and provides a new array. When filtering anything, it will return the information that you are looking for. For example, if you were only looking for even numbers in an array that possessed numeric values, you may call on the filter signature as such: let even = names.filter by the function “(Promineo Tech 08:30–09:21). Lastly, is the splice method which is very useful in altering the elements of an array by being able to add, modify or delete elements of a specific location in the array. The splice signature follows as let removeElement = homeworkDays.splice(parameter 1 location start, parameter 2 how many elements to remove). I could call for (1, 1) which would remove Wednesday as a homework day and only remove that one element and nothing before or after.

The favorite thing I learned this week is being able to go out and seek information about certain functions, methods, or arrays on my own and apply it to the lab and get the correct answer. There was certain material not provided in the videos and when I was practicing the lab, I had to figure out how to create the line of code with something I hadn’t been taught yet. I also realized in the moment that I hadn’t learned about it and what I would do to figure out how to get the result I was looking for. I am struggling to remember exactly what use every time I practice, but I am able to research and figure out what I need and apply it to my example.

Works Cited

Promineo Tech. “Intermediate Array Methods.” *YouTube*, uploaded by Promineo Tech, 15 July 2019, www.youtube.com/watch?v=5ywx-7wL840.