

SDI: Week 4

Additional Information

Activity: Project 4

In this activity, you're going to create a library of functions that solve some basic problems you may run into in your coding career. Solving them will also reinforce the concepts you've learning in the previous weeks and illustrate how those concepts can be used in concert to complete a task.

This project contains twelve (12) problems or tasks. You only need to solve six (6) of those twelve, and they can be any six that you choose.

Some strategies to consider:

- Don't wait until the last minute to get started. Some of these problems are easily solved while others will take more thought.
- Ask questions if you have them. I've tried to offer more full explanations of the requirements, but if you still have questions, let me know.
- Because this is a compilation of everything you have learned this term, **you cannot use regular expressions (RegEx) or jQuery**. Each of these tasks can be completed without the use of these additional resources.

Extra credit/Make-up Option:

Make-up points are available on this project. The eligibility guidelines are as follows:

- You must complete all twelve functions.
- You must submit your project on time.
- You must use comments within your code to document the sections that illustrate the concept for which you lost points. You also need to indicate which project it was.

Below are more robust descriptions of what is needed or expected for each function. Read them over carefully. If you still have questions, please feel free to email me.

String Functions

Does a string follow a 123-456-7890 pattern like a phone number?

This problem asks you to create a function that can accept a string and then determine if that string follows the pattern presented. For example, if you pass the string "123-45-67", this would not match the pattern. But, if you passed the string "407-695-0100", this would match the pattern. The only value returned from the function should be a Boolean; that is, true if the string matches the pattern or false if it does not.

Does a string follow an aaa@bbb.ccc pattern like an email address?

This problem is similar to the first except that the submitted string must follow a different pattern. For example, if you passed the string "fsosupport@fullsail", this would not match the pattern. But, if you passed the string "llewis@fullsail.com", this would match the pattern. The only value returned from the function should be a Boolean; that is, true if the string matches the pattern or false if it does not.

Is the string a URL (Does it start with http:// or https://)?

This task also involves looking at a submitted string to see if it includes the data necessary to indicate it is a URL. Thus, you should look for the http:// or https://. You do not have to worry if the rest of the URL is valid at this time. The only value returned from the function should be a Boolean; that is, true if the string matches the pattern or false if it does not.

Title-case a string (split into words, then uppercase the first letter of each word).

This problem involves sending a string into the function, changing the first letter of each word to uppercase while making sure the rest of the word is lowercase, and then returning the string from the function so it can be output.

Given a string that is a list of things separated by a given string, as well as another string separator, return a string with the first separator changed to the second: "a,b,c" + "," + "/" --> "a/b/c".

For this problem, you will need to send three arguments into your function. The first is the string with items separated by a separator (such as a comma). The second is the separator that is being replaced. The third parameter is the separator you are going to use in place of the original. Thus, if you send the parameters ("a,b,c", ",", "/"), the function should return the string as "a/b/c".

Number Functions

Format a number to use a specific number of decimal places as for money: 2.1 --> 2.10.

This function will require two parameters. The first is the number you wish to change, and the second is the number of decimal places you wish to use to format the number. For example, to format pi to three decimal places, you might send the following argument parameters: (3.14159, 3). The function will then return 3.142.

Fuzzy-match a number: is the number above or below a number within a certain percent?

This function confuses students the most, so please ask if you have questions. You will send three numbers as parameters for the function. You will then compare the first number to the second number to see if the first number is greater than or less than the second number, and then you will use the third number to see if the first number is within that percentage of the second number. So, if you were to send (5, 10, 50), the function would first determine if 5 is greater than or less than 10. It would then determine if 5 is within 50% of 10. The function should then return the results.

Find the number of hours or days difference between two dates.

This function requires you to send three parameters. The first two will be dates that you want to compare, and the third will be a string to determine if the function returns hours or days. The function will then perform the math necessary to find the difference between the two dates and return the choice of hours or days.

Given a string version of a number, such as "42", return the value as an actual Number data type, such as 42.

As you have learned, there is a difference between a number as a string data type and a number as a Number data type. For this task, you will simply need to send a string into the function and return it as a Number. If you want to get creative, you also can add a conditional to determine if the string sent into the function is a number before you do the conversion.

Array Functions

Find the smallest value in an array than is greater than a given number.

You will need to send two items into the function. The first is an array of numbers and the second will be a number you'll compare to numbers within the array. You will then return the number in the array that is the next highest number to the one you're using in the comparison. So, if you send the array [1,4,7,9,10,14,15] and the number 12 into the function, it should return 14 as the next highest number.

Find the total value of just the numbers in an array, even if some of the items are not numbers.

This one can be tricky if your array includes a string that is a number. You will send an array into the function and have it add together the numbers in the array. For example, if you send the array [1, "pickles", 3, "onions", 5, "10", 6, "SDI"], the function should return 15. Therefore, it should be able to recognize "10" as a string and ignore it from the total.

Given an array of objects and the name of a key, return the array sorted by the value of that key in each of the objects: "a" + [{a:2},{a:3},{a:1}] --> [{a:1},{a:2},{a:3}].

This function also requires you to send two argument parameters. The first one is an array of objects and the second one is a key. The function will then sort the array of objects by the key you send. In this case, you might send an array of objects [{a:2},{b:3},{a:1},{a:4}] and the key "a" which will then be sorted by the function using the key "a" and returned as [{a:1},{a:2},{a:3},{a:4},{b:3}].