

JavaScript ES6 Exercise

Exercise 1: variables, spread and rest

- ☐ Get an input 3 times (any type of input) using `prompt()`. Store the values like this:
 - `var firstInput = prompt()`
 - `let secondInput = prompt()`
 - `const thirdInput = prompt()`
- ☐ Store them in an array named `arr`.
- ☐ Using the spread operator, copy the values into a new array, `newArr`, with element 1 at the start, and 5 at the end. (i.e. `newArr == [1, firstInput, secondInput, thirdInput, 5]`).
- ☐ Using the rest operator, create a function, `restFcn`, which takes a first input parameter `newArr`, then the other input parameters will be the 1, inputs given and 5. (i.e. `restFcn(newArr, 1, firstInput, secondInput, thirdInput, 5)`). Store the rest of the values in `restOfValues`.
- ☐ In `restFcn`, compare the values of `newArr` and the rest of the inputs (gathered in `restOfValues` through the rest operator)
- ☐ During the loop, print the variables in `newArr` and `restOfValues`.
- ☐ Print "correct" if they are the same, "wrong" otherwise.

Exercise 2: functions

- ☐ Create three `const` arrays:
 - `arrA = [10,11,12,13,14,15]`
 - `arrB = [5,5,6,6,7,7]`
 - `quotients = []`
- ☐ Create an arrow function, `div`, with inputs `a`, `b`, `primeA = false`.
- ☐ This function shall divide `a` by `b` if `primeA == false`. Append the quotients to the `quotient` array.
- ☐ Create a `for()` loop to perform element-wise division between elements of `arrA` and `arrB`. Do the division using the function `div` where like so: `div(arrA[i], arrB[i])` or `div(arrA[i], arrB[i], true)`
- ☐ In the `for()` loop, when calling `div`, only put in the third input value `true` when `arrA[i]` is prime. You may brute-force this (instead of checking for primes).
- ☐ Print the `quotients` array.

Exercise 3: classes

- ☐ Create a class called `grades`.
- ☐ Have this class have the attributes: `math`, `history`, `science`, `health`, `coding`

- ☐ Have this class have the function *computeGWA* where it computes the average of the values of the *grades*' attribute values.
- ☐ Create a variable *studentGrades* with attribute values:
 - Math = 95
 - History = 75
 - Science = 89
 - Health = 45
 - Coding = 100
- ☐ Print the GWA computed.