

Lab 5 - Part 1: An Average Function

REQUIREMENTS: Using the provided code...

1. Create a function that takes 5 parameters.
2. This function will return the average of the five numbers it receives (to one decimal place). NB: Don't worry about validating the input.
3. Test your new function by outputting to the console with the values: 5, 10, 15, 20, 25. The result should be 15.0.
4. Now, you will use your function to determine your program average.
5. Create five variables (one for each course in this semester) and initialize them with a grade out of 100.
6. Using your custom function, create the logic that will send a success popup message if the overall average is 65 or more.
7. Create the logic that will send a "Review required" popup message if the average is below 65. Again, don't worry about validating the input.
8. Does your logic work? Make sure to test it with known values!

Lab 5 - Part 2: I Object!

REQUIREMENTS: Using the provided code...

1. Create a JavaScript object named ***meObject*** to represent yourself.
2. It will have four properties (you decide).
3. Use the console to output one of those properties.
4. Create a popup that will output two of those properties concatenated together. e.g. "My name is Sean and I am a teacher."

5. Comment the above line out and turn it into a method of your object.
6. Call this method.

Lab 5 - Part 3: Make the Bank

REQUIREMENTS: Using the provided code...

1. Create a JavaScript object to represent a bank customer.
2. Properties are: lastName, branchNumber, accountBalance, & interestRate (as a decimal value).
3. Methods are: makeDeposit, makeWithdrawal, & addInterest. addInterest should simply multiply the current balance - don't worry about compound interest.
4. Each method will return a string message: "Thank you, your current balance is now \$X.XX" with the updated balance to two decimal places.
5. Next, create the logic using your object to make a deposit transaction, then a withdrawal transaction, then a transaction that adds interest.
6. The string returned should be output to the console after each transaction.

EXTRA CHALLENGE: Add another property: multipleAccounts. This will hold a Boolean value. If set to *true*, the addInterest method will add .005 to the interest rate.