Created Objects Lab

Now that we know how to create objects using a constructor function (basically JavaScript's version of a class), let's see if we can incorporate some best practices into our roman numeral kata.

The details of how you take these steps are up to you and your partner. But you should continue pairing with each partner starting with a failing test and a simple requirement (red), making the test pass (green), then refactoring and finally picking up a new simple requirement and writing a failing test for the other partner to solve.

Your requirements
Find where you've created your romanNumeral object and change its simple declaration into a function that can be instantiated with the "new" keyword. Change it to a "RomanNumeral" function. This function/class will represent a single roman numeral.

- It should have a private property called "romanNumeralString" (Frankly, you can call it whatever you want as long as it represents the roman number representation of that number).
- romanNumeralString should have a getter and a setter function.
- RomanNumeral should also have a private property called "decimalNumber" which represents the deciman representation of the current number.
- decimalNumber will also with its own accessor functions.

When either setter is called, it should validate that the input parameter is valid for that type of number and then convert it to its output type. It should not return anything.

When the getter is called, it should simply return the proper value.

Alrighty then. Get after it! Enjoy!