**COMM644  
Assignment 4**

**Practice with JavaScript OOP Concepts (20 points)**

Within a new web page, create the following applications. Remember to comment out every application once you’ve finished with that step. Be sure to place all of your JavaScript code within an external script file:

1. Create a namespace called Animal. Then, define two sub classes from the namespace, one called Cat and another called Dog.
2. Abandon the use of namespaces and start over. Now, create two classes, one called Cat and another called Dog. The Cat class should be created using literal notation and the Dog class should be created using constructor notation.
3. Directly underneath, create a new instance of the Dog class.
4. Start over and now create a new class using constructor notation called Animal. When a new instance of the Animal class is created, the message “The Animal has been created” should be displayed in the console window.
5. Now, change the above code so that it accepts a parameter and that value is what should be displayed in the console window. The message should be passed as an argument into the constructor at the moment that the Animal class is instantiated.
6. Start over and now create a new class using constructor notation called Animal. Define five properties within your class including type, breed, color, height, and length. These properties will be set in the constructor so you’ll need to pass in parameters into the function’s constructor, set the properties, and then pass in the actual arguments during the object’s instantiation.
7. Use a for-in loop to loop through and display all of the properties in the Animal class. You should see a total of 5 properties displayed in a list within the console window.
8. Now, create a public method called speak. Within the speak method you will use a conditional to check the type of Animal being created. If it’s a dog, return “The <color> dog is barking!” If it’s a cat return “The <color> cat is meowing!” instead. Call that method after the Animal class has been instantiated.
9. Now, convert all of your properties to private properties. Then, create a private method called checkType(). In this method you will check to see what the type is. If it’s dog, return dog, otherwise, return cat. Then, create a public method called speak that returns the value of the checkType() method. The console window should now display “The <animal type> has made a noise!”
10. Create your own String method called findWords() that inherits from the native String Object. This method should find all instances of a specific word within a provided paragraph of text. It should then alert out to the user the number of times that word was found in the paragraph.

**The Recipe Card (15 points)**

Create an object to hold information on your favorite recipe. It should have properties for title (a string), servings (a number), and ingredients (an array of strings). Your object should have a publically available method that when called, will log out the recipe within the console so that the recipe information looks like this:

Guacamole  
Serves: 4  
Ingredients:   
- 3 Avocados  
- 1 Lime  
- 1 Teaspoon Salt  
- 1/2 Cup Onion  
- 3 Tablespoons Cilantro  
- 2 Diced Tomatoes  
- 1 Teaspoon Garlic  
- 1 Pinch Ground Pepper

**The Reading List (15 points)**

Keep track of which books you read and which books you want to read! Create an array of objects, where each object describes a book and has properties for the title (a string), author (a string), and alreadyRead (a boolean indicating if you read it yet). Iterate through the array of books. For each book, log the book title and book author like so: "The Hobbit by J.R.R. Tolkien". Then, use a conditional to change the output depending on whether you read it yet or not. If you read it, log a string like 'You already read "The Hobbit" by J.R.R. Tolkien', and if not, log a string like 'You still need to read "The Lord of the Rings" by J.R.R. Tolkien.' Make sure your object contains at least 5 books.