# Week 4 – JavaScript: String Functions and Regular Expressions

## Overview

This week, we explore advanced string operations and dive into the world of regular expressions (RegEx), a powerful tool for pattern matching and data validation. These exercises will help you process and validate textual data, a critical skill in real-world web applications.

**Videos to Watch Before Starting**

1. [JavaScript String Methods Explained](https://www.youtube.com/watch?v=8Ey3nZ7zvpY)
2. [Introduction to Regular Expressions](https://www.youtube.com/watch?v=rhzKDrUiJVk)
3. [Validating Data with RegEx](https://www.youtube.com/watch?v=EkluES9Rvak)

**Practice Exercises**

**1. Filtering Words in an Array**

A messaging app needs to filter messages containing the word "hello" (case-insensitive).

**Task**:  
Write a function getHello(array) that filters and logs all strings in the input array containing the word "hello". Use methods like .filter() and regular expressions.

Example Input:  
["bye", "hello123", "newhello", "he20llo", "hello", "abchello", "xyzabc"]

Expected Output:  
hello123, newhello, he20llo, hello, abchello

**2. Replacing Numbers in a String**

A text processor needs to replace all occurrences of "10" with "Ten".

**Task**:  
Write a function replaceTen(str) that replaces all instances of "10" in a given string using a regular expression.

Example Input:  
"There are 10 people in room number 10. Call all of the 10 people outside."

Expected Output:  
"There are Ten people in room number Ten. Call all of the Ten people outside."

**3. Cleaning Up Text Formatting**

An editor tool needs to standardize text formatting based on specific rules.

**Task**:  
Write a function cleanUp(str) that applies the following corrections to the input string:

* Replace double spaces with single spaces after punctuation (. ? !).
* Convert double quotes to single quotes.
* Remove unnecessary spaces inside parentheses.

Example Input:  
"This is a sentence. This is another."  
"A ( red ) dog arrived."

Expected Output:  
"This is a sentence. This is another."  
"A (red) dog arrived."

**4. Validating and Formatting Canadian Postal Codes**

An e-commerce platform needs to clean and validate Canadian postal codes.

**Task**:  
Write a function fixPostalCode(postalCode) that:

* Removes leading/trailing spaces.
* Converts letters to uppercase.
* Ensures the fourth character is a space.
* Validates the postal code format (A1A 1A1).

Throw an error for invalid postal codes.

Example Input:  
"m5w1e6"

Expected Output:  
"M5W 1E6"

**5. Mapping Postal Codes to Provinces**

Using postal codes, identify the province they belong to.

**Task**:  
Write a function toProvince(postalCode, useLongForm) that:

1. Uses fixPostalCode to validate and format the postal code.
2. Maps the first letter of the postal code to a province.
3. Returns the province name in short form (default) or long form (useLongForm = true).

Example Input:  
"M5W 1E6"

Expected Output:  
"ON" (default) or "Ontario" (long form)

**6. Checking the First Character**

A form validator needs to ensure that names start with an uppercase letter.

**Task**:  
Write a function checkFirstChar(str) that checks if the first character is uppercase. Use a ternary operator for optimization.

Example Input:  
"Keyin"

Expected Output:  
"String's first character is uppercase"

**7. Validating Email Addresses**

A web application needs to validate email formats.

**Task**:  
Write a function validEmail(email) that checks if the input string is a valid email using a regular expression. Ensure the email adheres to these rules:

* Contains uppercase/lowercase letters, digits, and special characters.
* The period (.) cannot start, end, or repeat consecutively.

Example Input:  
"example@keyincollege.com"

Expected Output:  
true

**8. Trimming Strings Without .trim()**

Write a custom implementation of the .trim() method.

**Task**:  
Write a function myTrimFunction(str) that removes leading and trailing spaces from a string using regular expressions.

Example Input:  
" Keyin College "

Expected Output:  
"Keyin College"

**9. Validating HTML Tags**

An HTML validator needs to check for valid tags.

**Task**:  
Write a function validateHTML(tag) that validates if the input is a proper HTML tag (e.g., <b>, <html>).

Example Input:  
"<b>"

Expected Output:  
true

**Research Assignment**

Research and discuss the following:

1. How regular expressions are implemented in JavaScript.
2. The differences between test(), match(), and replace() methods.