**Mini Linter**

In this project, you will use what you know about iterating over arrays to improve the quality of a paragraph and gather some information about that paragraph.

This is the same type of work that word processing software does. Additionally, you may have heard of *linting*, a process by which text is evaluated and improved by an application. In this project, you will create a miniature version of a linter using array methods that you have learned.

let story = 'Last weekend, I took literally the most beautiful bike ride of my life. The route is called "The 9W to Nyack" and it actually stretches all the way from Riverside Park in Manhattan to South Nyack, New Jersey. It\'s really an adventure from beginning to end! It is a 48 mile loop and it basically took me an entire day. I stopped at Riverbank State Park to take some extremely artsy photos. It was a short stop, though, because I had a really long way left to go. After a quick photo op at the very popular Little Red Lighthouse, I began my trek across the George Washington Bridge into New Jersey. The GW is actually very long - 4,760 feet! I was already very tired by the time I got to the other side. An hour later, I reached Greenbrook Nature Sanctuary, an extremely beautiful park along the coast of the Hudson. Something that was very surprising to me was that near the end of the route you actually cross back into New York! At this point, you are very close to the end.';

let overusedWords = ['really', 'very', 'basically'];

let unnecessaryWords = ['extremely', 'literally', 'actually' ];

let storyWords = story.split(' ');

/// ^^^^^ set initial variables

/// log original story word count

console.log('The original story contains ' + storyWords.length + ' words\n');

/// remove unnecessary words

let betterWords = storyWords.filter(word => !unnecessaryWords.includes(word));

console.log('The story contains ' + betterWords.length + ' words after removing unnecessary words\n');

/// count and log over used words

overusedWords.forEach(function(overWord) {

let overWordCount = 0;

betterWords.filter(function(storyWord) {

if (overWord === storyWord) {

overWordCount++;

}

});

if (overWordCount != 1) {

console.log('The Word ' + overWord.toUpperCase() + ' was used ' + overWordCount + ' times\n');

} else {

console.log('The Word ' + overWord.toUpperCase() + ' was used ' + overWordCount + ' time\n');

}

});

/// count sentences

let sentenceCount = 0;

betterWords.forEach(word => {

if (word[word.length-1] === '.' || word[word.length-1] === '!') {

sentenceCount++;

}

});

console.log('There are ' + (sentenceCount + 1) + ' sentences in the story\n');

/// log the refined story

console.log(betterWords.join(' ') + '\n');

///REMOVE AND REPLACE EVERY OTHER INSTANCE OF OVER USED WORDS

console.log('----------------------------------------------------');

console.log('\nProject: Remove and replace every other instance of overused words!\n\n');

///Create an array to hold a count for each item in 'overusedWords'

let overusedWC = overusedWords.map(function() {

return 0;

});

///Initiate a count for removed words

let removedWC = 0;

///Filter out every other instance of the contents of 'overUsedWords'

let lessOverusedWords = betterWords.map(function(storyWord) {

if (!overusedWords.includes(storyWord)) {

return storyWord;

} else {

let pos = overusedWords.indexOf(storyWord);

if (overusedWC[pos] < 1) {

overusedWC[pos]++;

return storyWord;

} else { ///Add alternative word

overusedWC[pos] = 0;

removedWC++;

///Here we could pull from an array of alternative words

return 'REMOVED';

}

}

});

/// Log results

console.log(lessOverusedWords.join(' ') + '\n')

console.log(removedWC + ' overused words were removed \n');

overusedWords.forEach(function(overWord) {

let overWordCount = 0;

lessOverusedWords.filter(function(storyWord) {

if (overWord === storyWord) {

overWordCount++;

}

});

if (overWordCount != 1) {

console.log('The Word ' + overWord.toUpperCase() + ' was used ' + overWordCount + ' times\n');

} else {

console.log('The Word ' + overWord.toUpperCase() + ' was used ' + overWordCount + ' time\n');

}

});

/// FIND THE WORD THAT APEARS THE GREATEST NUMBER OF TIMES

console.log('----------------------------------------------------')

console.log('\nProject: Find the word that appears the greatest number of time! \n\n')

/// Remove puntuation and capitalisation

/// Create an array of accepted characters to be used to filter out punctuation

const acceptedChars = [' ', 'a', 'b', 'c', 'd', 'e', 'f', 'g', 'h', 'i', 'j', 'k', 'l', 'm', 'n', 'o', 'p', 'q', 'r', 's', 't', 'u', 'v', 'w', 'x', 'y', 'z', '0', '1', '2', '3', '4', '5', '6', '7', '8', '9'];

/// Replace capitals with lower case

let noCaps = story.toLowerCase();

/// Create an array of individual characters

let noCapsChars = noCaps.split('');

/// Filter out punctuation characters

let noPunctChars = noCapsChars.filter(char => acceptedChars.includes(char));

/// Recreate story with no capitals or punctuation

let noPunctStory = noPunctChars.join('');

/// Replace double spaces with single spaces to eliminate blank entries in word arrays

let noPunctSentences = noPunctStory.split(' ');

let noDoubleSpaces = noPunctSentences.join(' ');

/// Create an array of story words

let noPunctWords = noDoubleSpaces.split(' ');

/// Create an array of individual words

let noDupes = [];

noPunctWords.forEach(function(word) {

if (!noDupes.includes(word)) {

noDupes.push(word);

}

});

/// Count how many instances of each individual word

let instanceCount =[];

noDupes.forEach(function(countWord) {

let count = 0;

noPunctWords.forEach(function(storyWord) {

if (countWord === storyWord) {

count++;

}

});

/// Create a 2D array to store each word and its count

instanceCount.push([count, countWord]);

});

/// Sort the array by decending word count

instanceCount.sort(function(a, b) {

return b[0] - a[0];

});

/// Log the most used words

console.log('The Most Used Words. \n'.toUpperCase())

for (let i = 0; instanceCount[i][0] > 3; i++) {

console.log('The word ' + instanceCount[i][1].toUpperCase() + ' was used ' + instanceCount[i][0] + ' times')

}