Question 1.

function fizzBuzz() {

for (let i = 1; i <= 100; i++) {

if (i % 3 === 0 && i % 5 === 0) {

console.log("FizzBuzz");

}

else if (i % 3 === 0) {

console.log("Fizz");

}

else if (i % 5 === 0) {

console.log("Buzz");

}

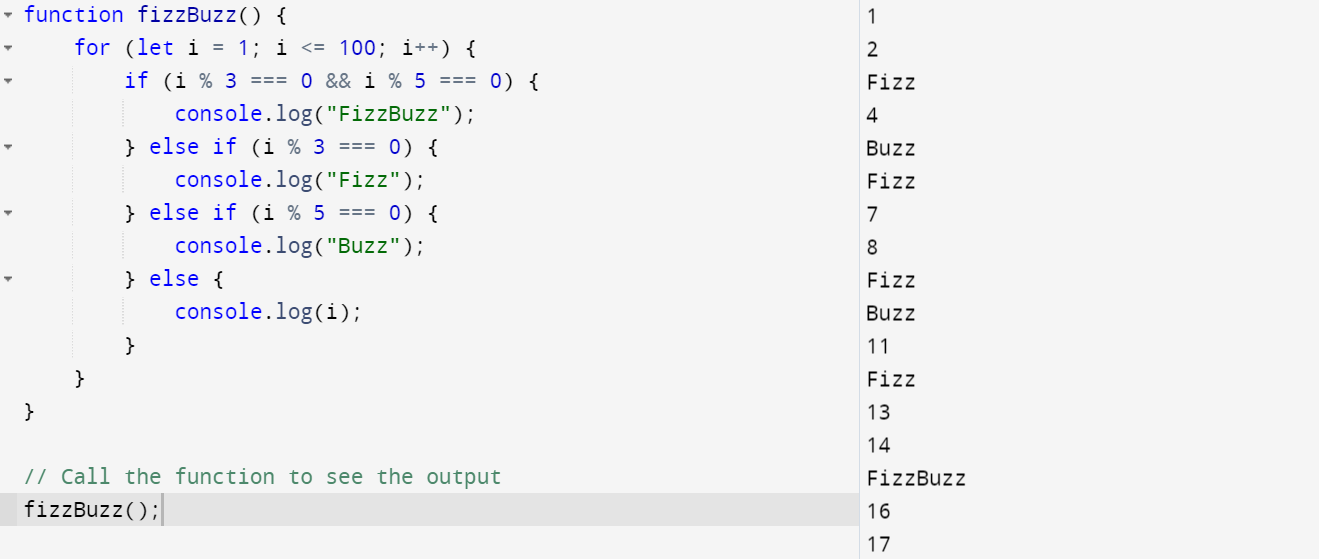
else { console.log(i);

}

}

}

// Call the function to see the output

fizzBuzz();

Question 2.

function evaluateExpression(expression) {

// Remove any spaces from the expression

expression = expression.replace(/\s+/g, '');

// Use a regular expression to split the expression into numbers and operators

let numbers = expression.split(/[-+]/).map(Number);

let operators = expression.split(/[0-9]+/).filter(op => op);

// Initialize the result with the first number

let result = numbers[0];

// Iterate through the operators and numbers to compute the result

for (let i = 0; i < operators.length; i++) {

let operator = operators[i];

let number = numbers[i + 1];

if (operator === '+') {

result += number;

} else if (operator === '-') {

result -= number;

}

}

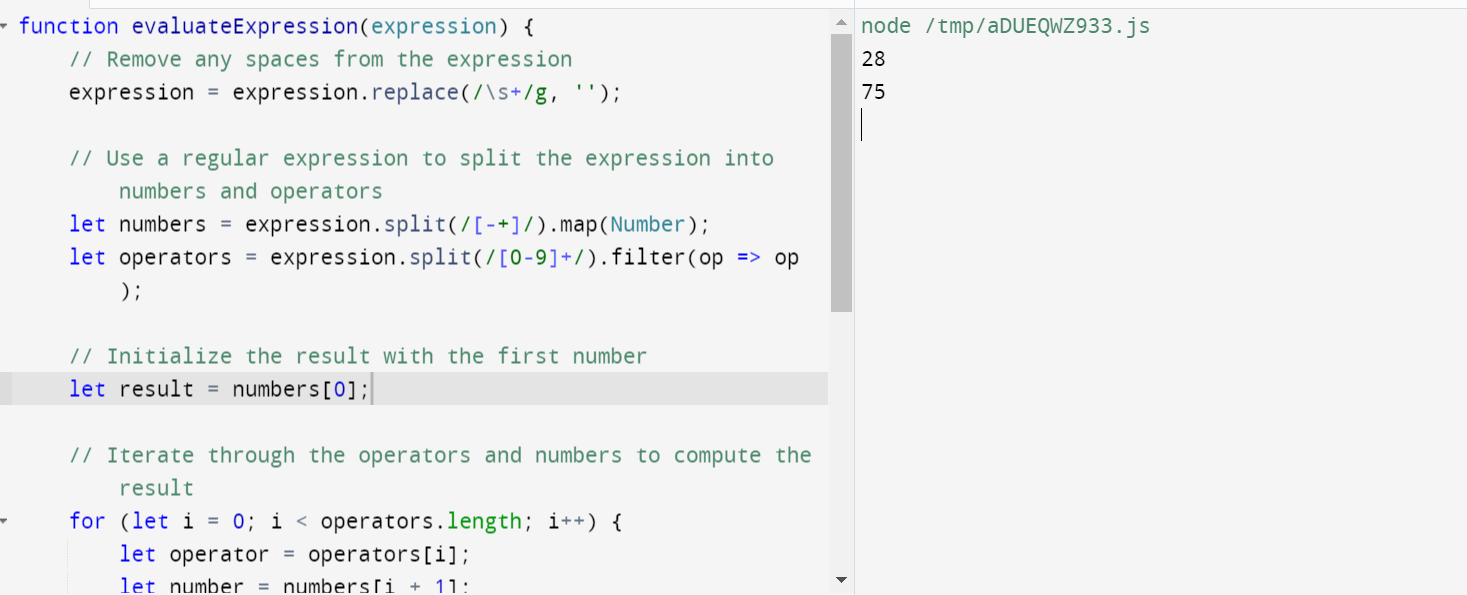
return result;

}

// Example usage

console.log(evaluateExpression("10 + 20 - 5 + 3")); // Output: 28

console.log(evaluateExpression("100 - 50 + 25")); // Output: 75



Question 3.

function flattenArray(nestedArray) {

let flattened = [];

nestedArray.forEach(element => {

if (Array.isArray(element)) {

flattened = flattened.concat(flattenArray(element));

} else {

flattened.push(element);

}

});

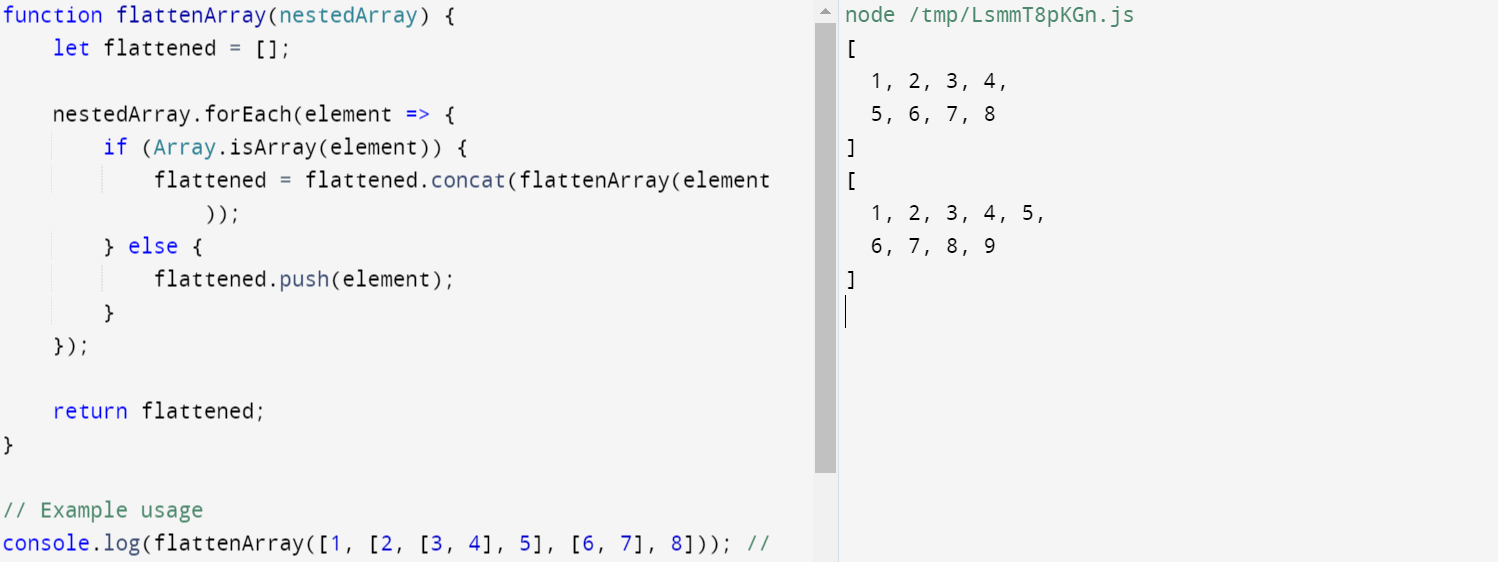
return flattened;

}

// Example usage

console.log(flattenArray([1, [2, [3, 4], 5], [6, 7], 8])); // Output: [1, 2, 3, 4, 5, 6, 7, 8]

console.log(flattenArray([1, [2, 3], [4, [5, 6], 7], 8, [9]])); // Output: [1, 2, 3, 4, 5, 6, 7, 8, 9]



Question 4.

function areAnagrams(str1, str2) {

// Helper function to sort the characters in a string

function sortString(str) {

return str.split('').sort().join('');

}

// Remove any non-alphabetic characters and convert to lowercase

str1 = str1.replace(/[^a-zA-Z]/g, '').toLowerCase();

str2 = str2.replace(/[^a-zA-Z]/g, '').toLowerCase();

// Compare the sorted versions of the strings

return sortString(str1) === sortString(str2);

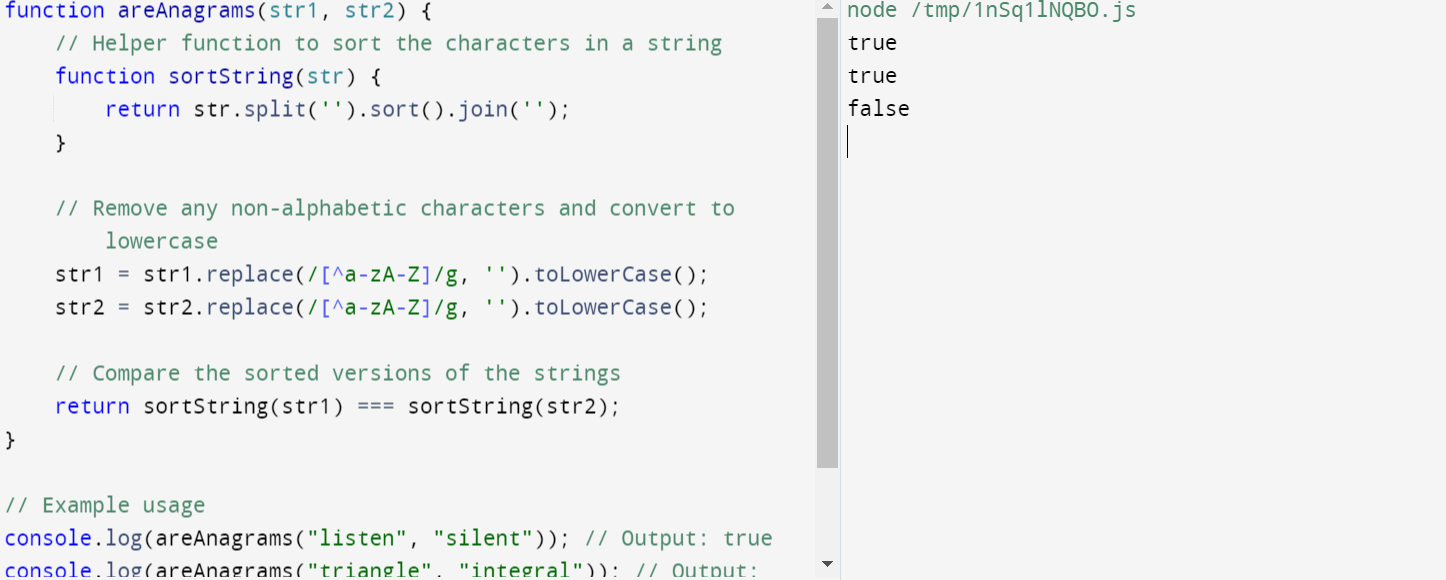
}

// Example usage

console.log(areAnagrams("listen", "silent")); // Output: true

console.log(areAnagrams("triangle", "integral")); // Output: true

console.log(areAnagrams("apple", "pale")); // Output: false



Question 5.

function removeDuplicates(array) {

// Use a Set to automatically handle duplicates

let uniqueSet = new Set(array);

// Convert the Set back to an array

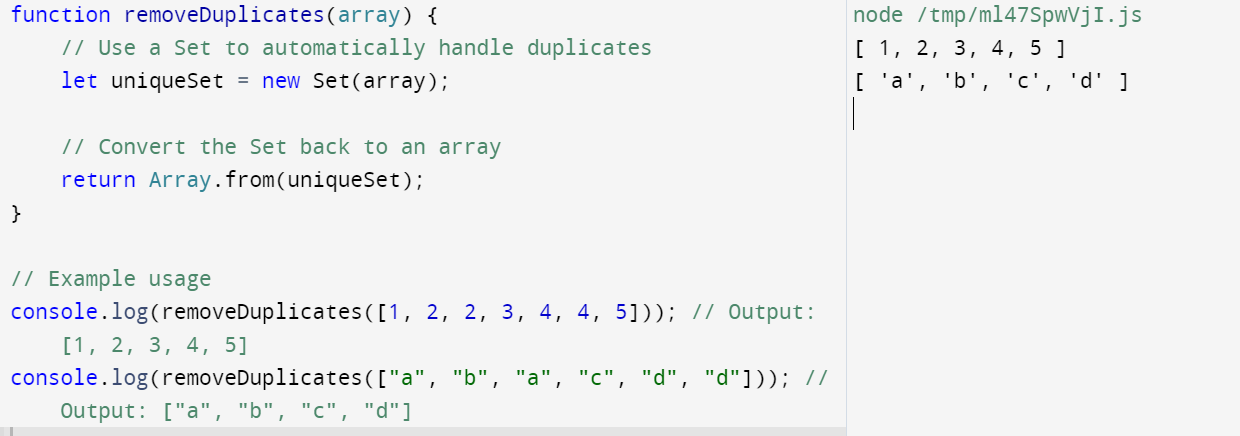
return Array.from(uniqueSet);

}

// Example usage

console.log(removeDuplicates([1, 2, 2, 3, 4, 4, 5])); // Output: [1, 2, 3, 4, 5]

console.log(removeDuplicates(["a", "b", "a", "c", "d", "d"])); // Output: ["a", "b", "c", "d"]



Question 6.

function capitalizeFirstLetter(str) {

return str.split(' ').map(word => {

return word.charAt(0).toUpperCase() + word.slice(1).toLowerCase();

}).join(' ');

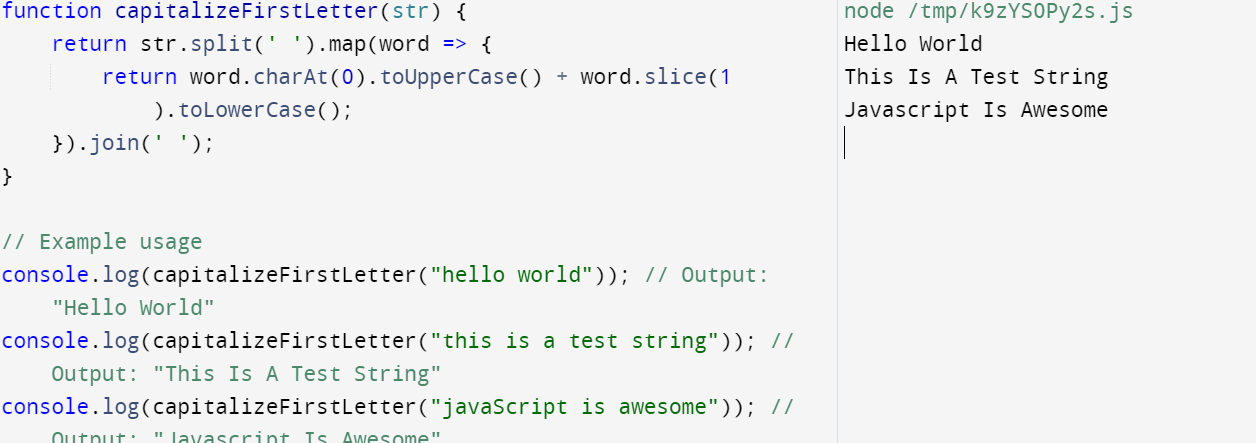
}

// Example usage

console.log(capitalizeFirstLetter("hello world")); // Output: "Hello World"

console.log(capitalizeFirstLetter("this is a test string")); // Output: "This Is A Test String"

console.log(capitalizeFirstLetter("javaScript is awesome")); // Output: "Javascript Is Awesome"



Question 7.

function fibonacci(n) {

let sequence = [];

// Handle edge cases

if (n <= 0) {

return sequence;

} else if (n === 1) {

sequence.push(0);

return sequence;

} else if (n >= 2) {

sequence.push(0, 1);

for (let i = 2; i < n; i++) {

sequence.push(sequence[i - 1] + sequence[i - 2]);

}

}

return sequence;

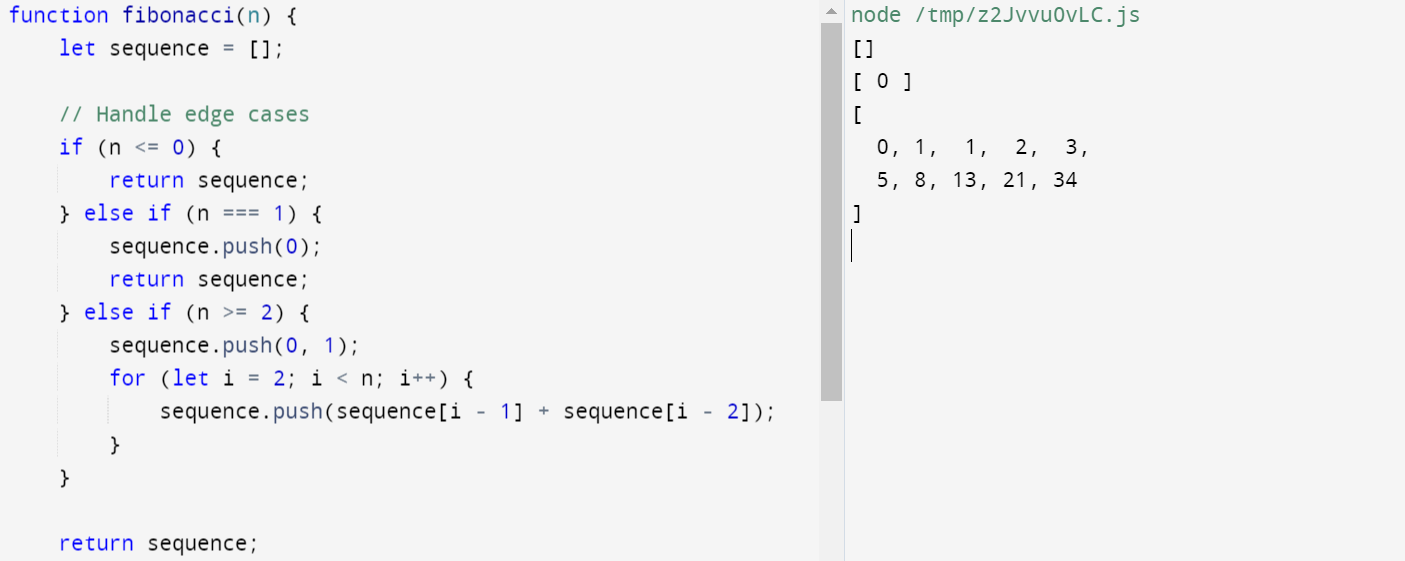
}

// Example usage

console.log(fibonacci(0)); // Output: []

console.log(fibonacci(1)); // Output: [0]

console.log(fibonacci(10)); // Output: [0, 1, 1, 2, 3, 5, 8, 13, 21, 34]



Question 8.

class HashMap {

constructor() {

this.map = {};

}

// Method to add or update a key-value pair

put(key, value) {

this.map[key] = value;

}

// Method to get the value associated with a key

get(key) {

return this.map[key];

}

// Method to remove a key-value pair

remove(key) {

if (this.map.hasOwnProperty(key)) {

delete this.map[key];

}

}

}

// Example usage

let map = new HashMap();

// Adding key-value pairs

map.put("apple", 5);

map.put("banana", 10);

map.put("cherry", 15);

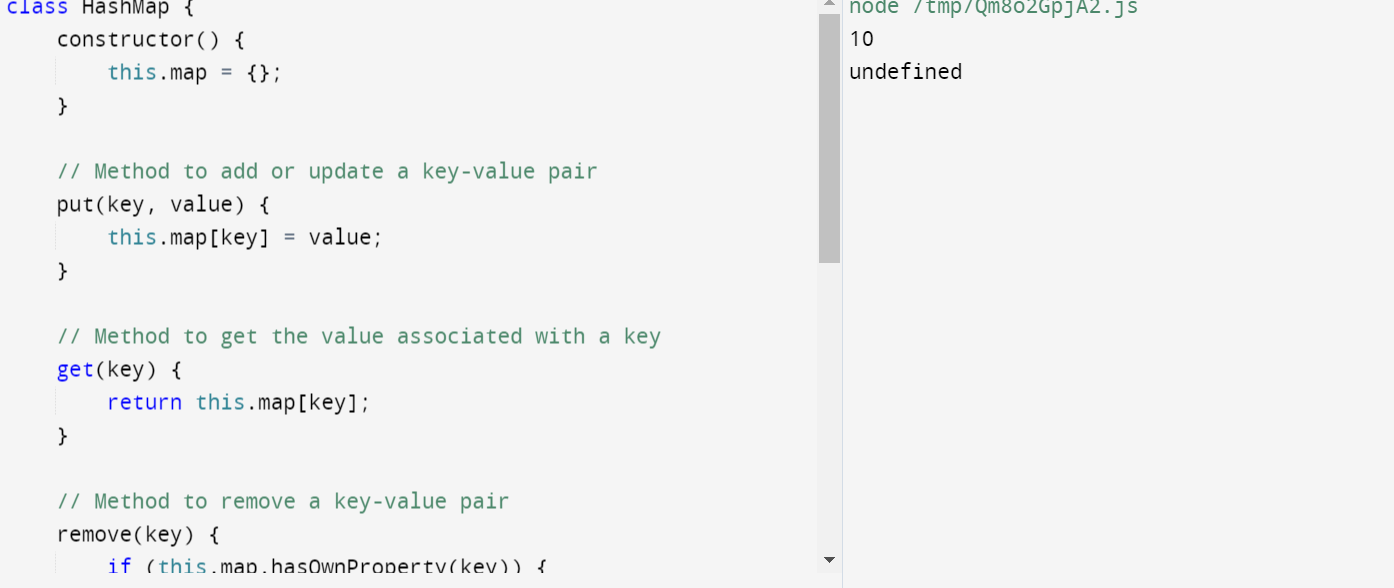
// Retrieving values

console.log(map.get("banana")); // Output: 10

// Removing a key-value pair

map.remove("banana");

console.log(map.get("banana")); // Output: undefined



Question 9.

function filterEvenNumbers(array) {

let result = [];

for (let i = 0; i < array.length; i++) {

if (array[i] % 2 !== 0) {

result.push(array[i]);

}

}

return result;

}

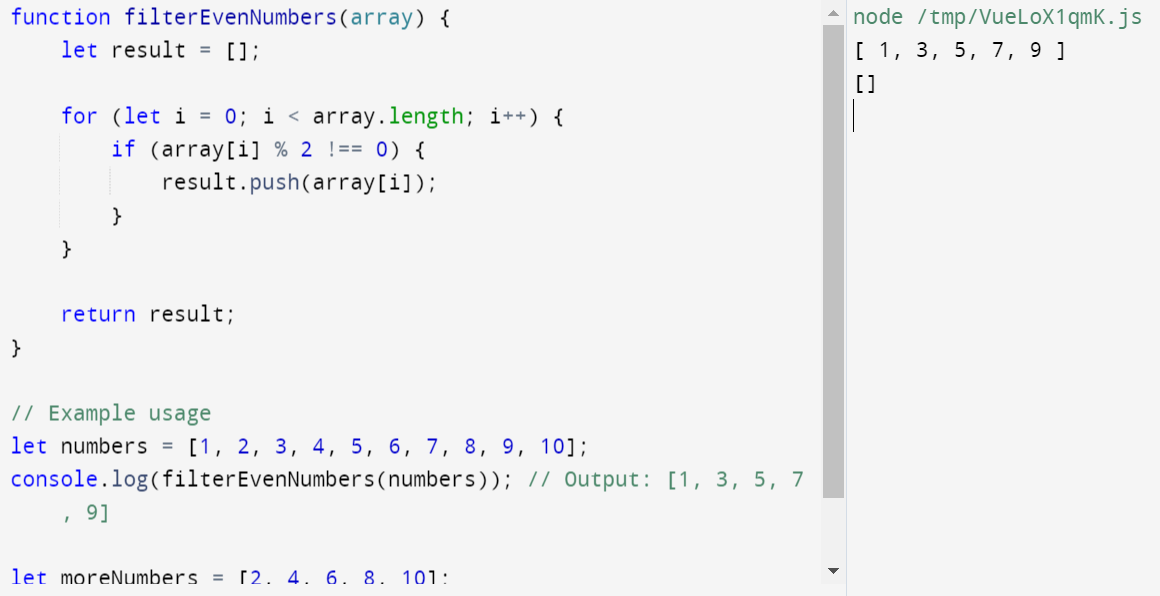
// Example usage

let numbers = [1, 2, 3, 4, 5, 6, 7, 8, 9, 10];

console.log(filterEvenNumbers(numbers)); // Output: [1, 3, 5, 7, 9]

let moreNumbers = [2, 4, 6, 8, 10];

console.log(filterEvenNumbers(moreNumbers)); // Output: []



Question 10 .

function toTitleCase(str) {

// Split the string into words

let words = str.split(' ');

// Capitalize the first letter of each word

for (let i = 0; i < words.length; i++) {

words[i] = words[i].charAt(0).toUpperCase() + words[i].slice(1).toLowerCase();

}

// Join the words back into a single string

return words.join(' ');

}

// Example usage

console.log(toTitleCase("hello world")); // Output: "Hello World"

console.log(toTitleCase("this is a test string")); // Output: "This Is A Test String"

console.log(toTitleCase("javaScript is awesome")); // Output: "Javascript Is Awesome"

