**Problem 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);**

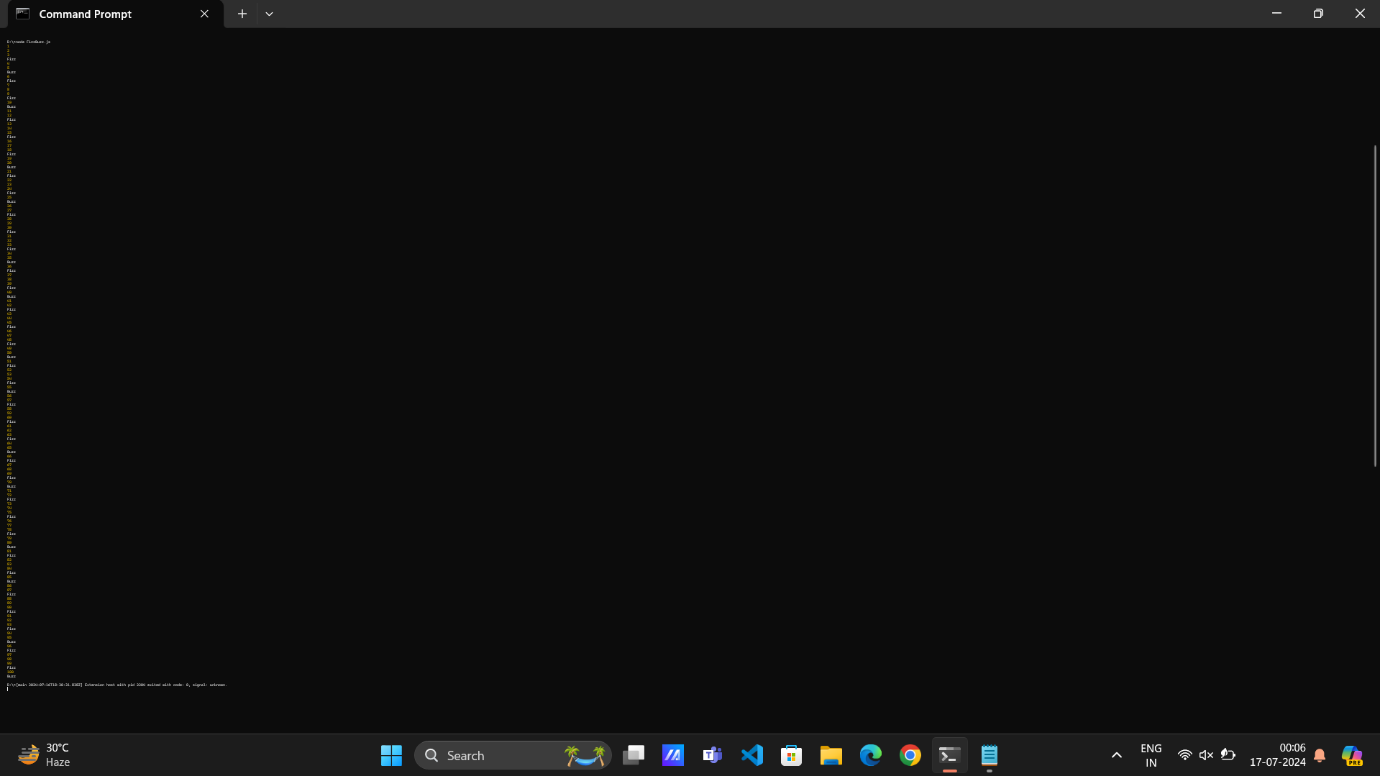
**}**

**}**

**}**

**fizzBuzz();**

**Output:**



**Problem 2:**

**function evaluateExpression(expression) {**

**let tokens = expression.match(/[-+]?([0-9]\*\.[0-9]+|[0-9]+)/g);**

**let result = 0;**

**let currentOperator = '+';**

**for (let token of tokens) {**

**switch (currentOperator) {**

**case '+':**

**result += parseFloat(token);**

**break;**

**case '-':**

**result -= parseFloat(token);**

**break;**

**}**

**let nextOperatorIndex = expression.indexOf(token) + token.length;**

**if (nextOperatorIndex < expression.length) {**

**currentOperator = expression[nextOperatorIndex];**

**}**

**}**

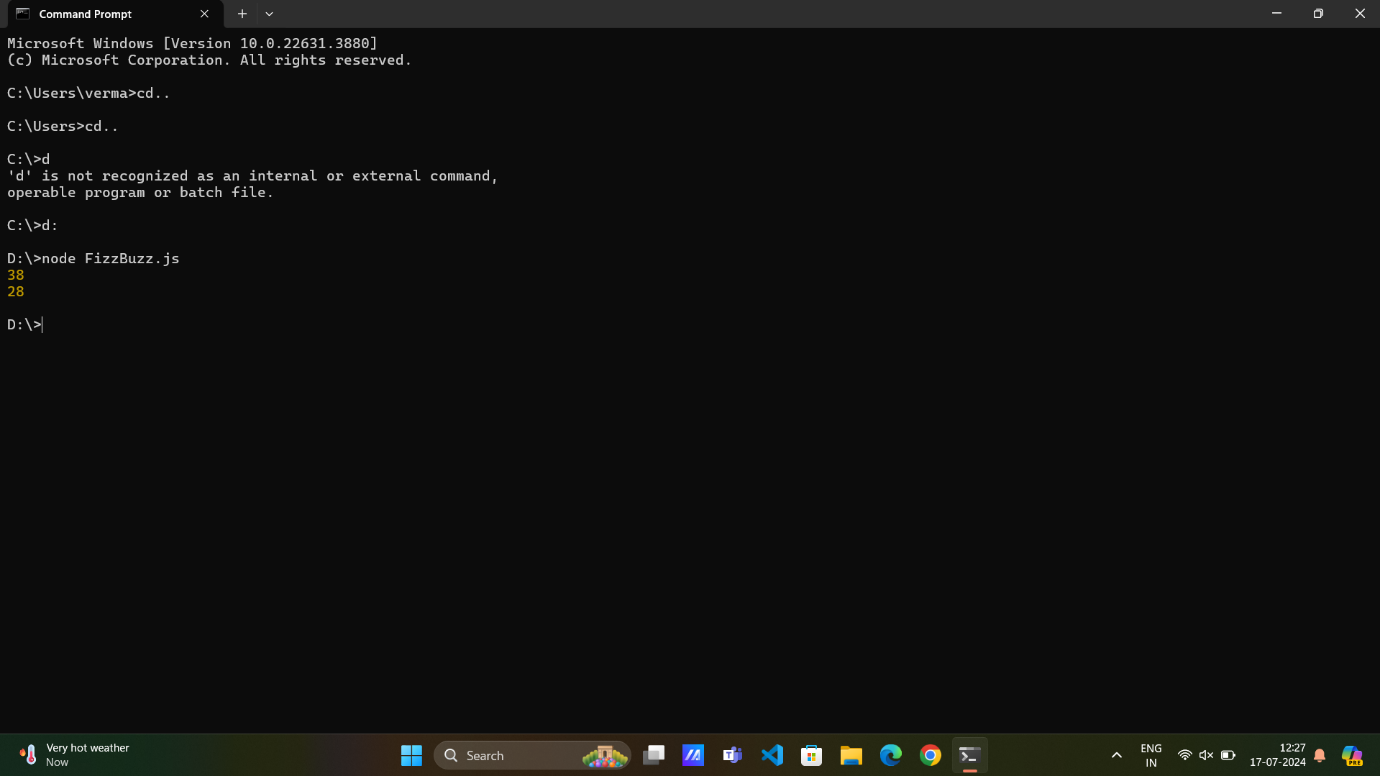
**return result;**

**}**

**console.log(evaluateExpression("10+20-5+3"));**

**console.log(evaluateExpression("15-3+8-2"));**

**Output:**

****

**Problem 3:**

**function flattenArray(nestedArray) {**

**let flatArray = [];**

**function flatten(arr) {**

**for (let element of arr) {**

**if (Array.isArray(element)) {**

**flatten(element);**

**} else {**

**flatArray.push(element);**

**}**

**}**

**}**

**flatten(nestedArray);**

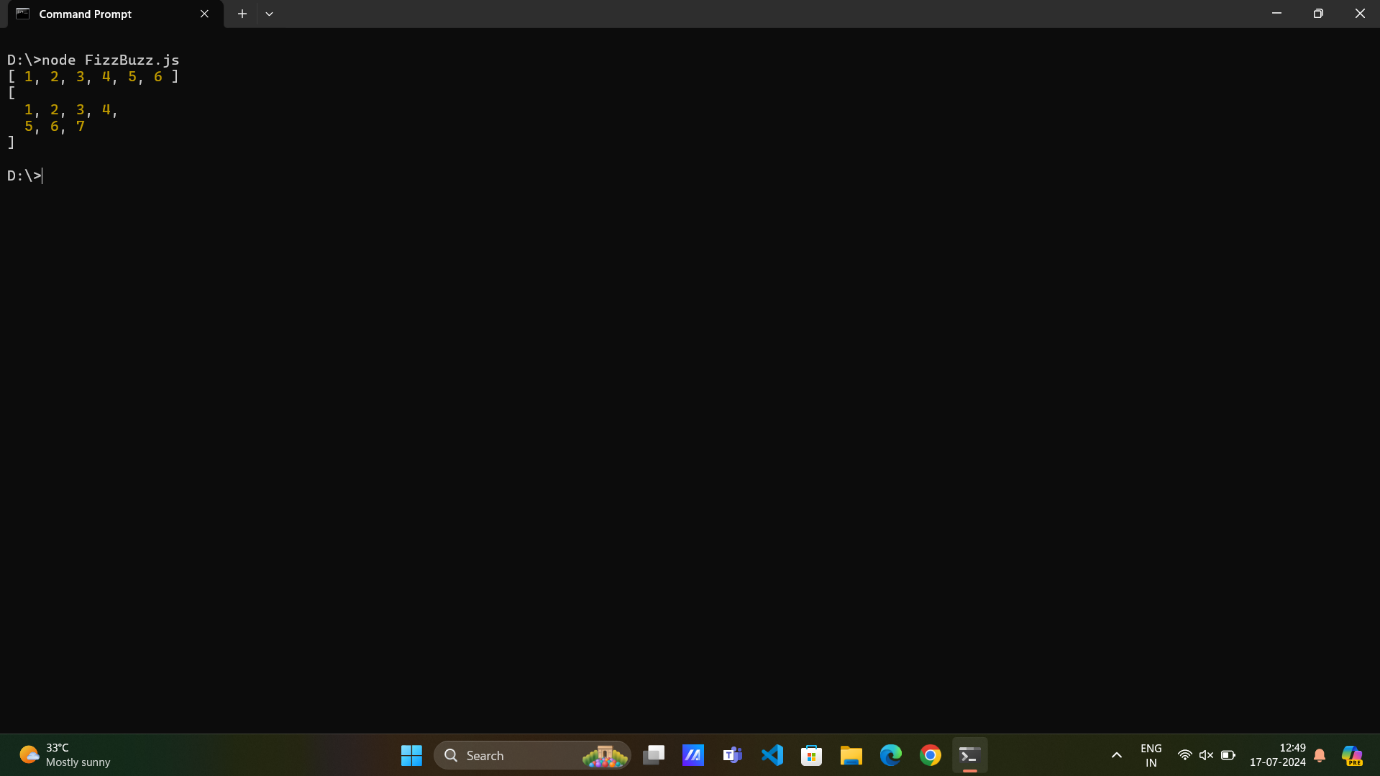
**return flatArray;**

**}**

**console.log(flattenArray([1, [2, [3, 4], 5], 6]));**

**console.log(flattenArray([[1, 2, [3]], 4, [5, [6, 7]]]));**

**Output:**

****

**Problem 4:**

**function areAnagrams(str1, str2) {**

**str1 = str1.replace(/[^\w]/g, '').toLowerCase();**

**str2 = str2.replace(/[^\w]/g, '').toLowerCase();**

**if (str1.length !== str2.length) {**

**return false;**

**}**

**let sortedStr1 = str1.split('').sort().join('');**

**let sortedStr2 = str2.split('').sort().join('');**

**return sortedStr1 === sortedStr2;**

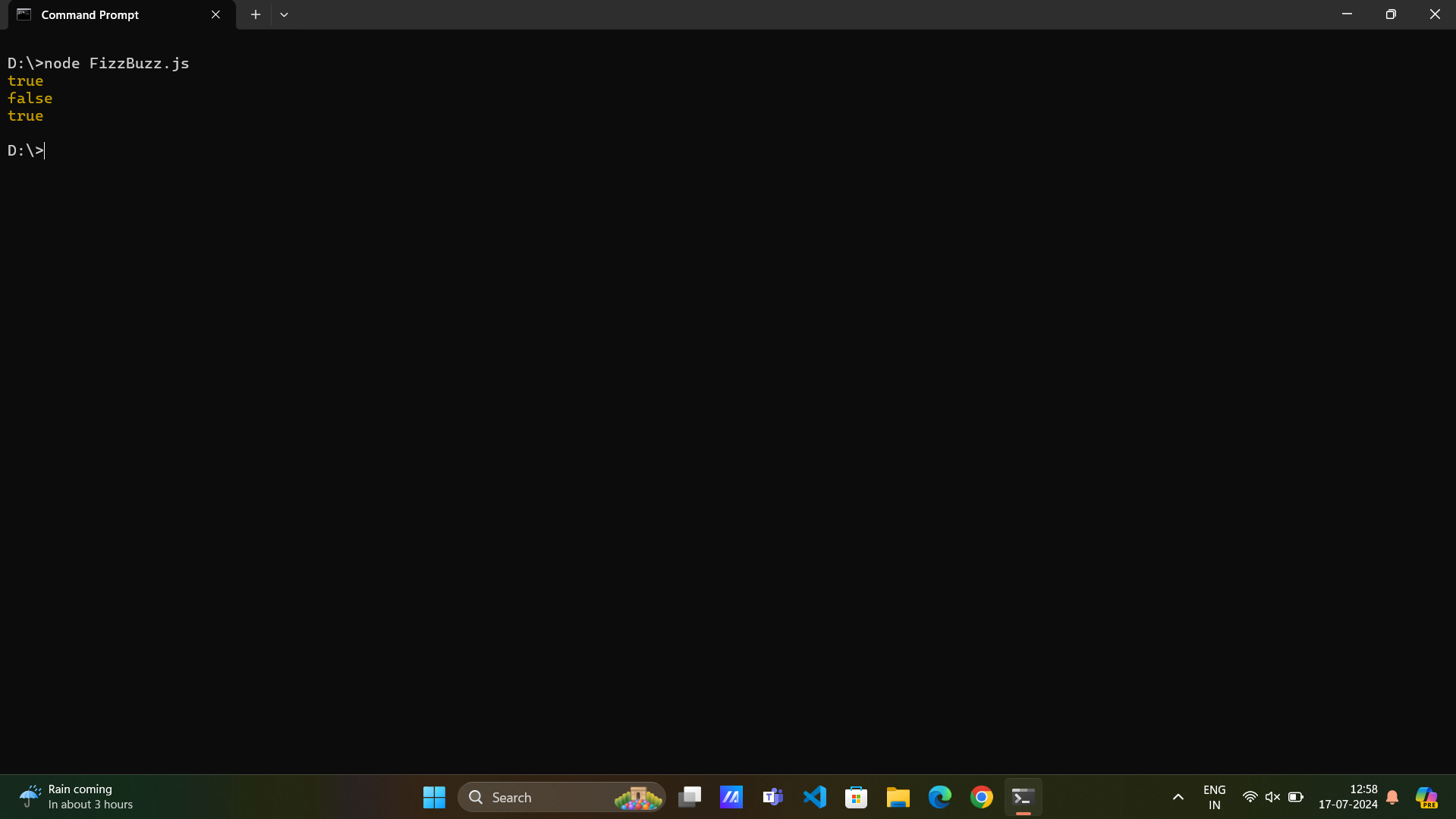
**}**

**console.log(areAnagrams("listen", "silent"));**

**console.log(areAnagrams("hello", "world"));**

**console.log(areAnagrams("Dormitory", "Dirty room!"));**

**Output:**

****

**Problem 5:**

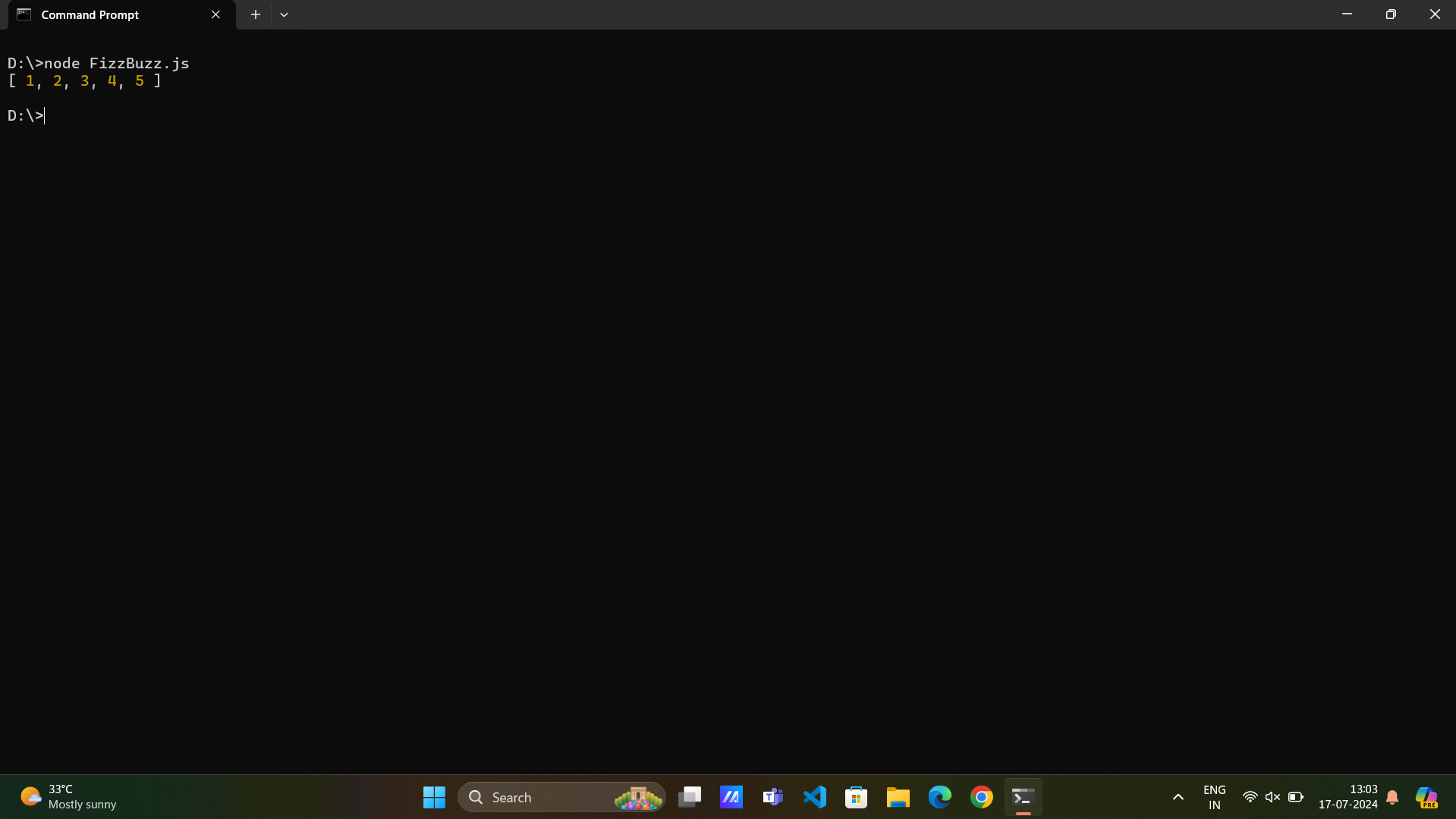
**function removeDuplicates(array) {**

**return Array.from(new Set(array));**

**}**

**console.log(removeDuplicates([1, 2, 2, 3, 4, 4, 5]));**

**Output:**

****

**Problem 6:**

**function capitalizeWords(str) {**

**let words = str.split(" ");**

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

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

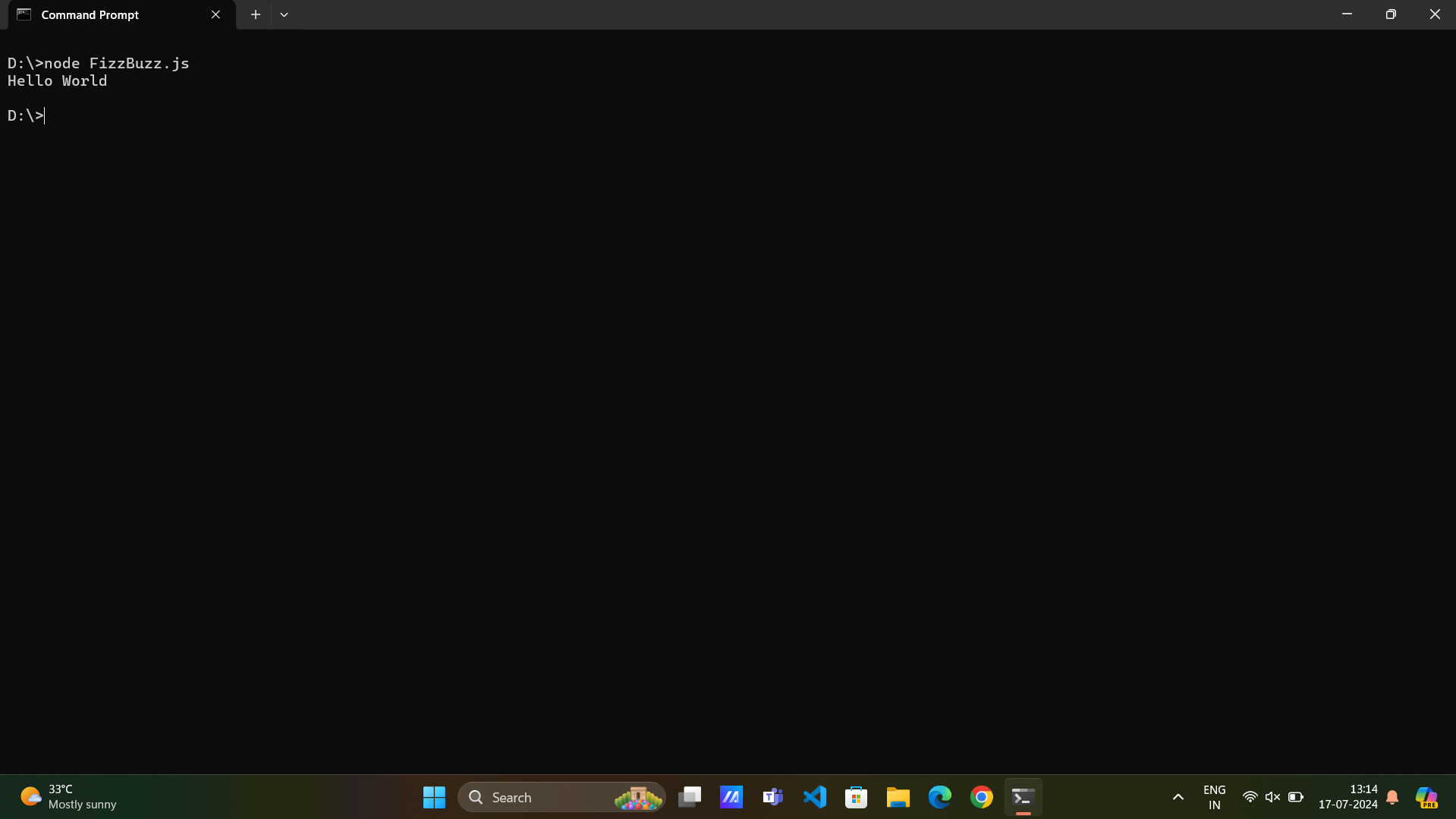
**}**

**return words.join(" ");**

**}**

**console.log(capitalizeWords("hello world"));**

**Output:**

****

**Problem 7:**

**function fibonacci(n) {**

**let sequence = [];**

**if (n <= 0) {**

**return sequence;**

**}**

**sequence.push(0);**

**if (n > 1) {**

**sequence.push(1);**

**}**

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

**let num = sequence[i - 1] + sequence[i - 2];**

**sequence.push(num);**

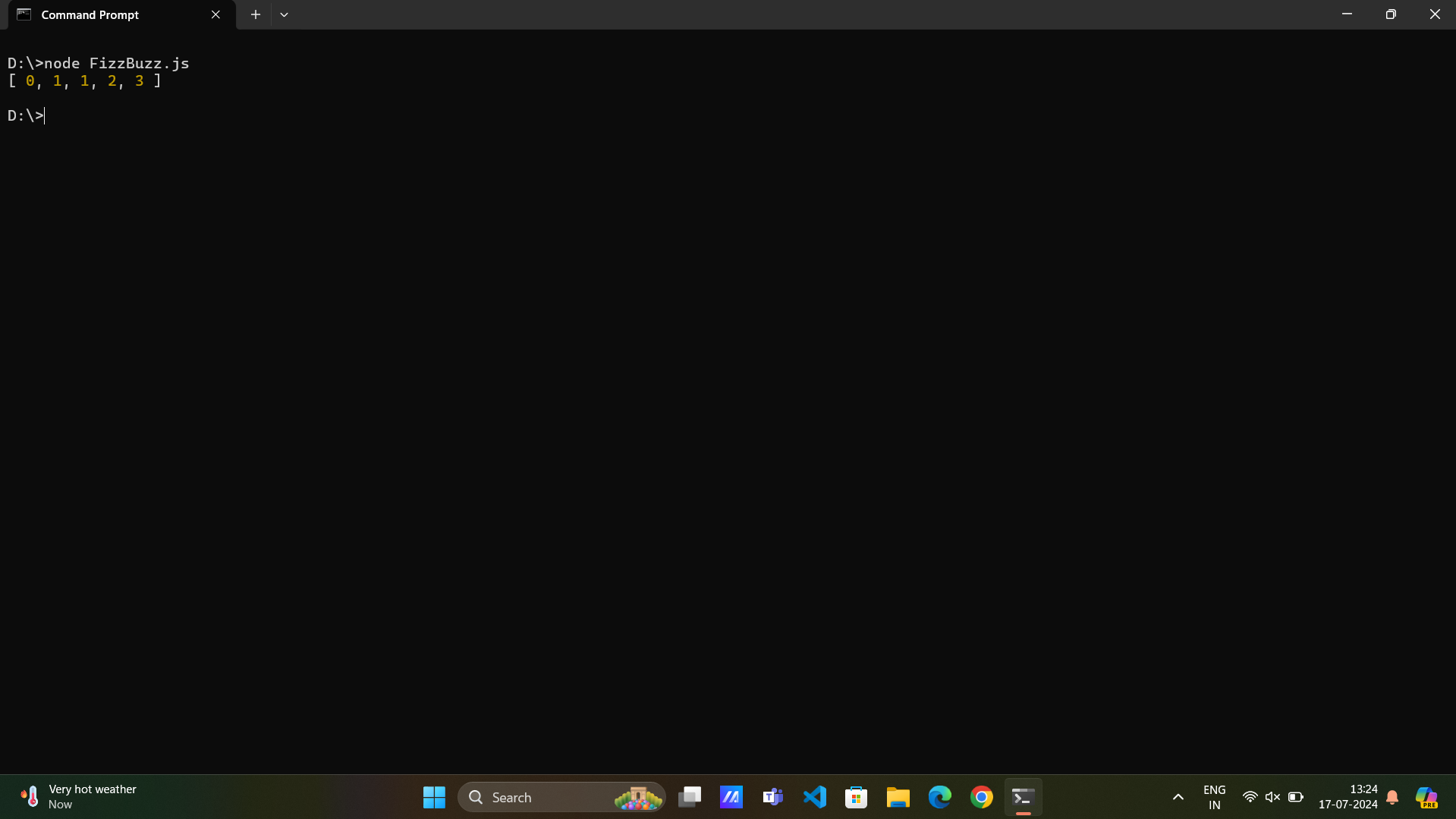
**}**

**return sequence;**

**}**

**console.log(fibonacci(5));**

**Output:**

****

**Problem 8:**

**class HashMap {**

**constructor() {**

**this.map = {};**

**}**

**put(key, value) {**

**this.map[key] = value;**

**}**

**get(key) {**

**return this.map[key];**

**}**

**remove(key) {**

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

**delete this.map[key];**

**}**

**}**

**}**

**let myMap = new HashMap();**

**myMap.put("name", "John");**

**myMap.put("age", 30);**

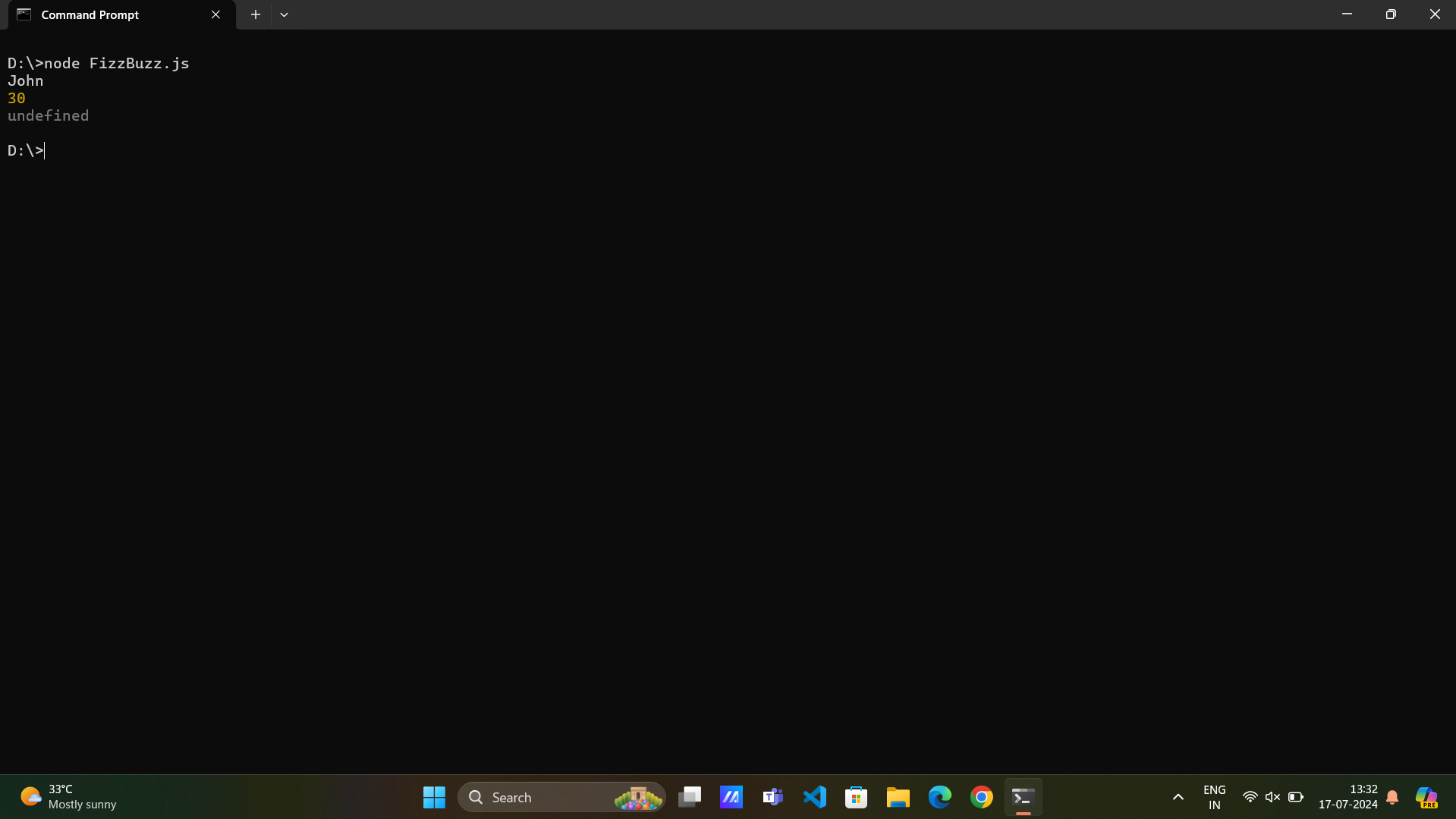
**console.log(myMap.get("name"));**

**console.log(myMap.get("age"));**

**myMap.remove("age");**

**console.log(myMap.get("age"));**

**Output:**

****

**Problem 9:**

**function filterOutEvenNumbers(array) {**

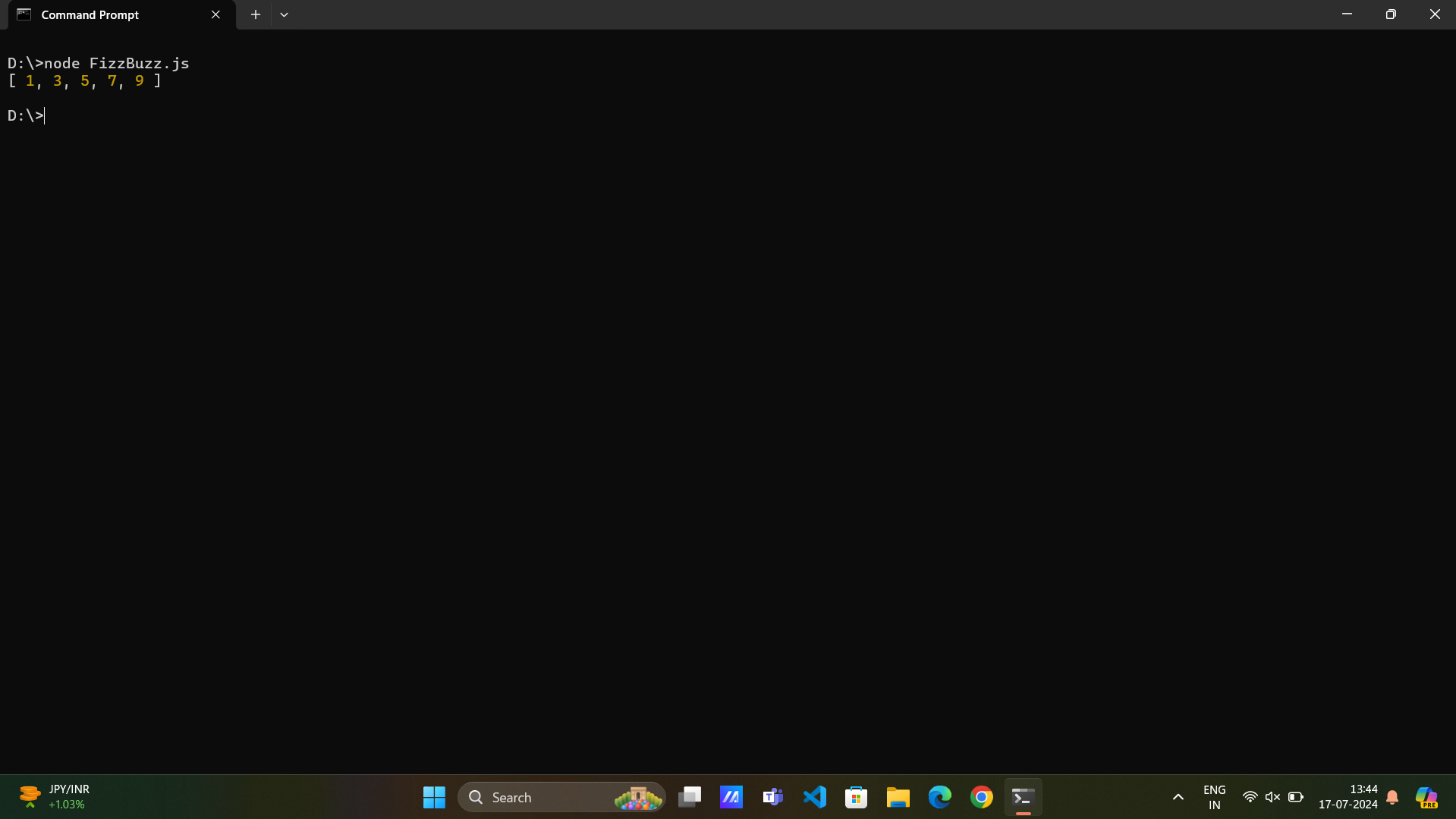
**return array.filter(num => num % 2 !== 0);**

**}**

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

**console.log(filterOutEvenNumbers(numbers));**

**Output:**

****

**Problem 10:**

**function toTitleCase(str) {**

**let words = str.split(" ");**

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

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

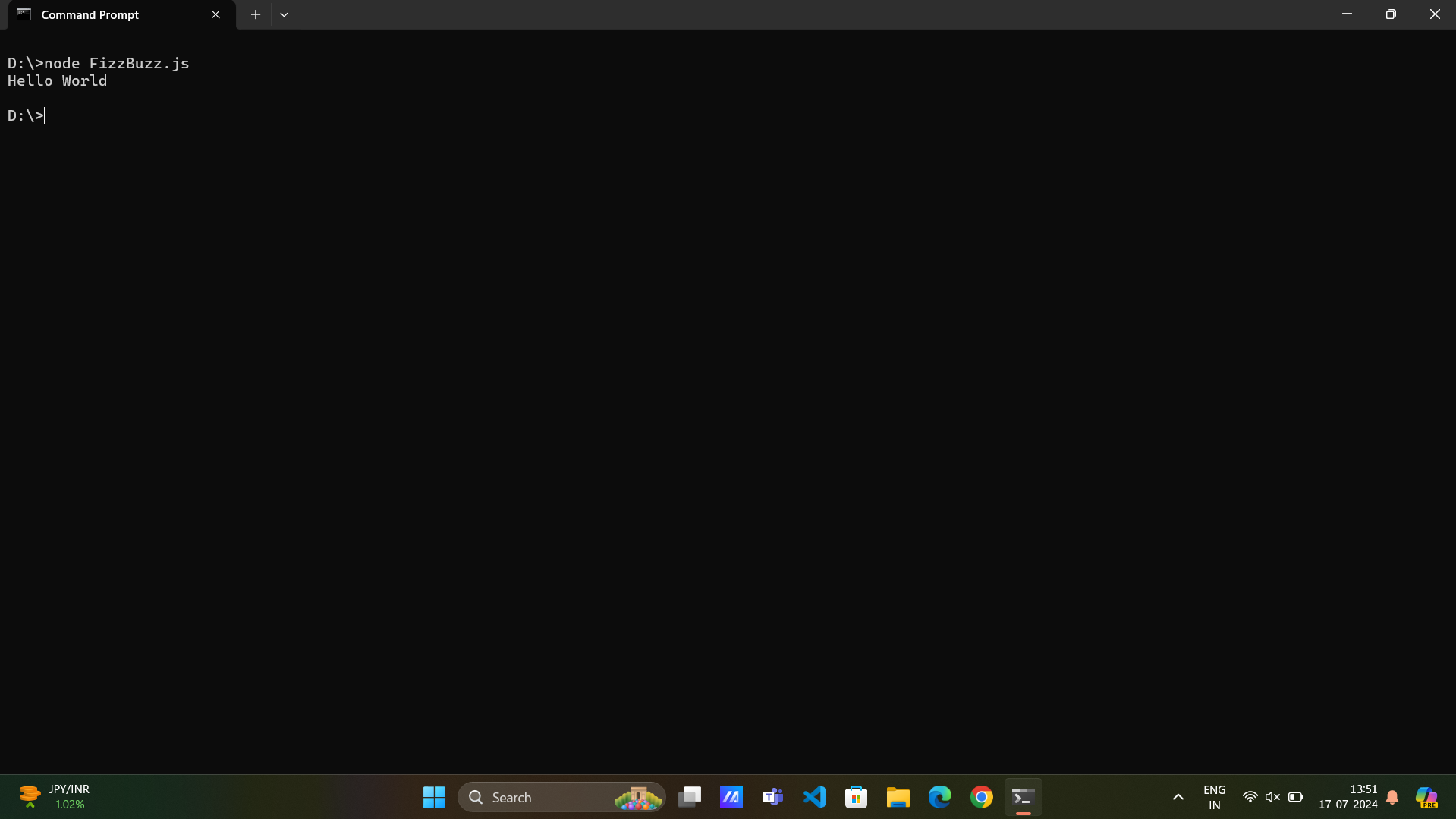
**}**

**return words.join(" ");**

**}**

**console.log(toTitleCase("hello world"));**

**Output:**

****