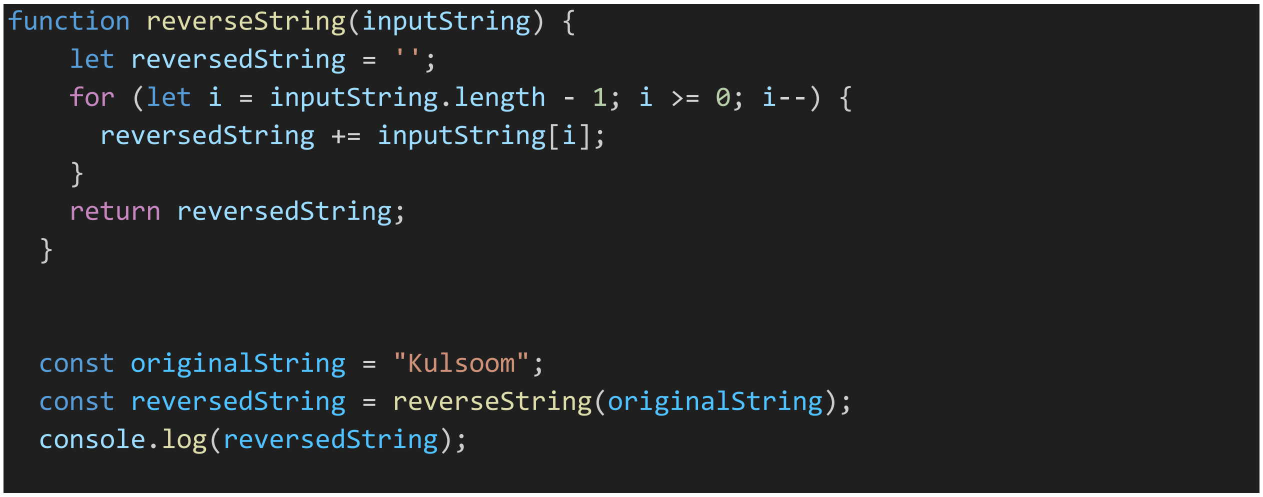
WD-II F (2-4)

Name: KULSOOM

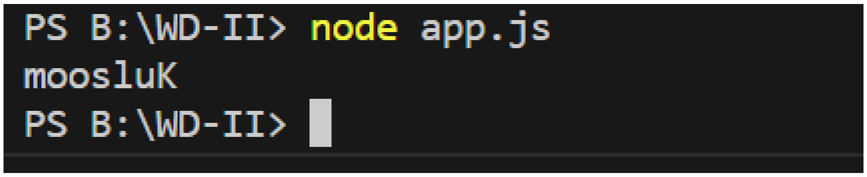
Contact #: 03407834231

1. Question: Reverse a string without using the built-in reverse() method.



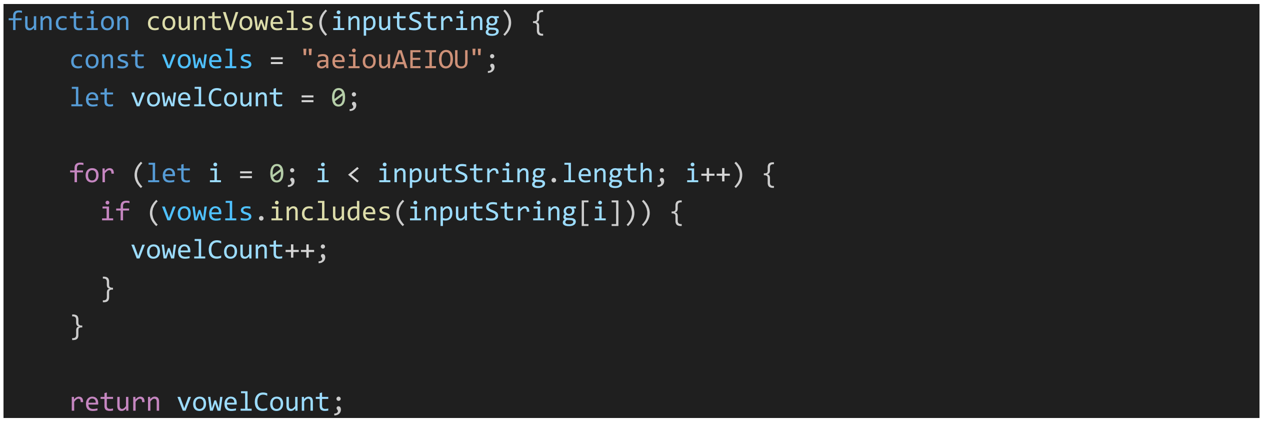
```
function reverseString(inputString) {
    let reversedString = '';
    for (let i = inputString.length - 1; i >= 0; i--) {
      reversedString += inputString[i];
    }
    return reversedString;
  }


  const originalString = "Kulsoom";
  const reversedString = reverseString(originalString);
  console.log(reversedString);
```

OUTPUT:

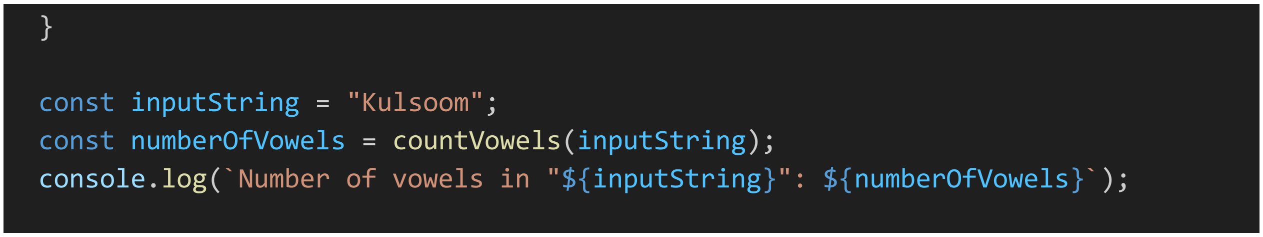


```
PS B:\WD-II> node app.js
moosluK
PS B:\WD-II> 
```

2. Question: Count the number of vowels in a given string.

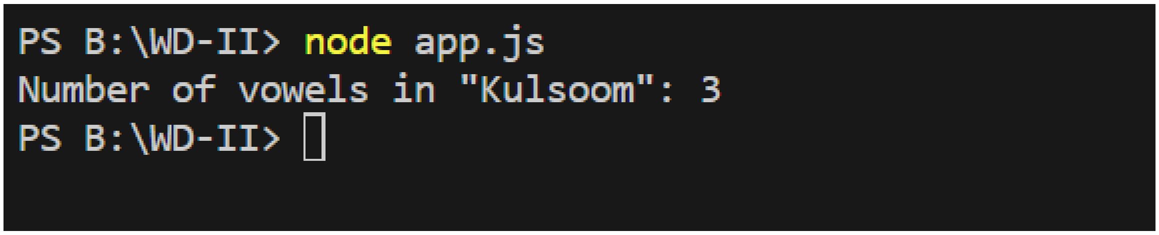


```
function countVowels(inputString) {
    const vowels = "aeiouAEIOU";
    let vowelCount = 0;

    for (let i = 0; i < inputString.length; i++) {
      if (vowels.includes(inputString[i])) {
        vowelCount++;
      }
    }

    return vowelCount;
```



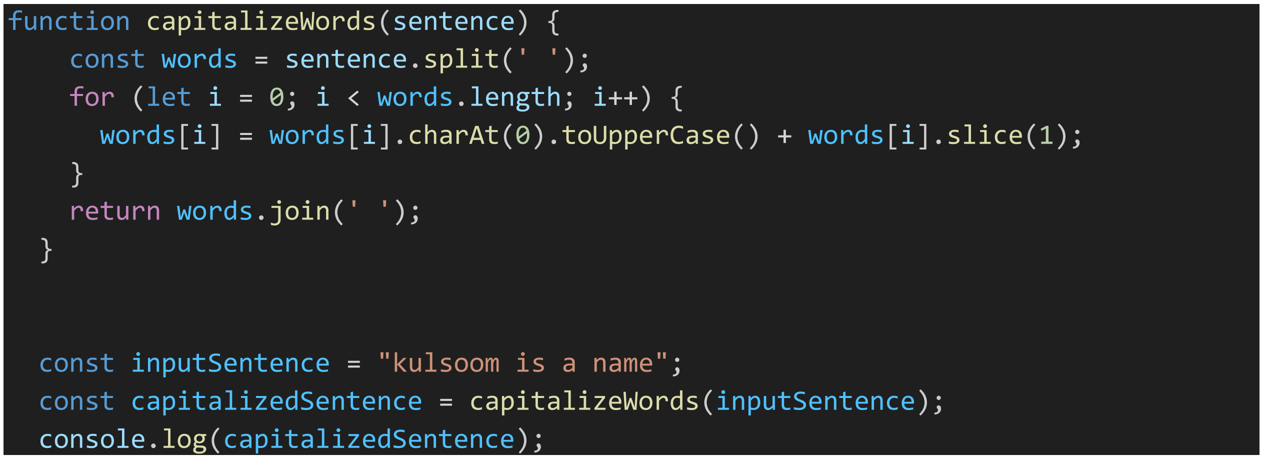
```
  }

  const inputString = "Kulsoom";
  const numberOfVowels = countVowels(inputString);
  console.log(`Number of vowels in "${inputString}": ${numberOfVowels}`);
```

OUTPUT:



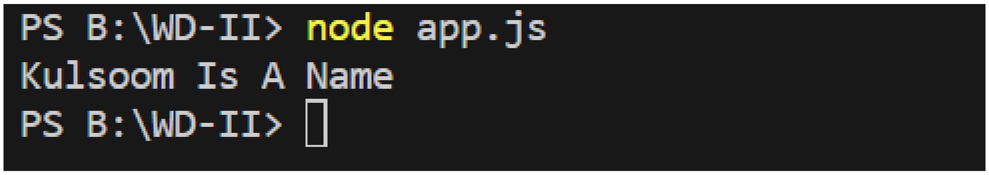
```
PS B:\WD-II> node app.js
Number of vowels in "Kulsoom": 3
PS B:\WD-II> 
```

3. Question: Convert the first letter of each word in a sentence to uppercase.



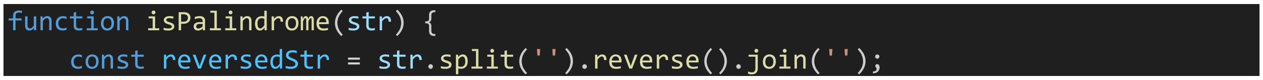
```
function capitalizeWords(sentence) {
    const words = sentence.split(' ');
    for (let i = 0; i < words.length; i++) {
      words[i] = words[i].charAt(0).toUpperCase() + words[i].slice(1);
    }
    return words.join(' ');
  }


  const inputSentence = "kulsoom is a name";
  const capitalizedSentence = capitalizeWords(inputSentence);
  console.log(capitalizedSentence);
```

OUTPUT:

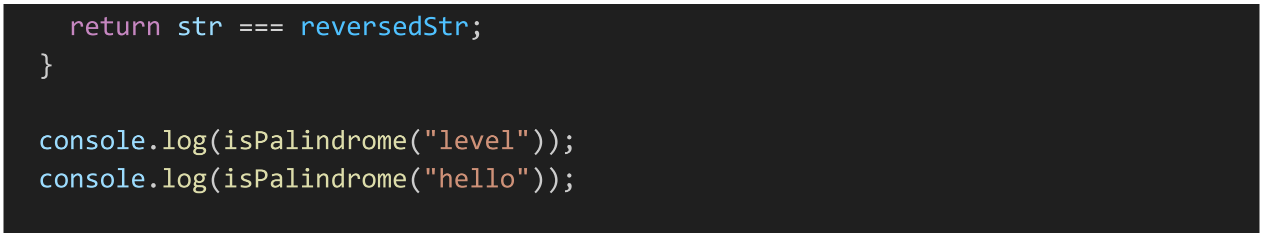


```
PS B:\WD-II> node app.js
Kulsoom Is A Name
PS B:\WD-II> 
```

4. Question: Check if a string is a palindrome.

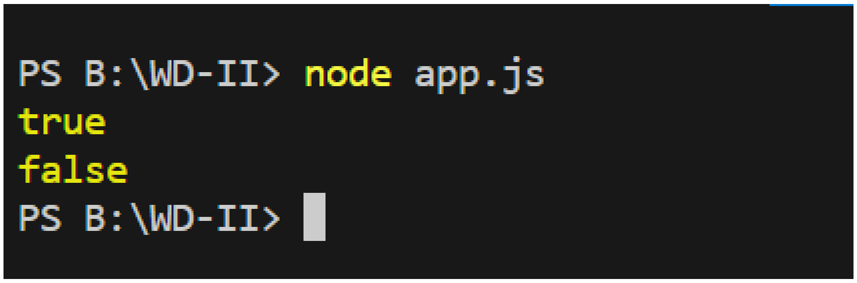


```
function isPalindrome(str) {
    const reversedStr = str.split('').reverse().join('');
```



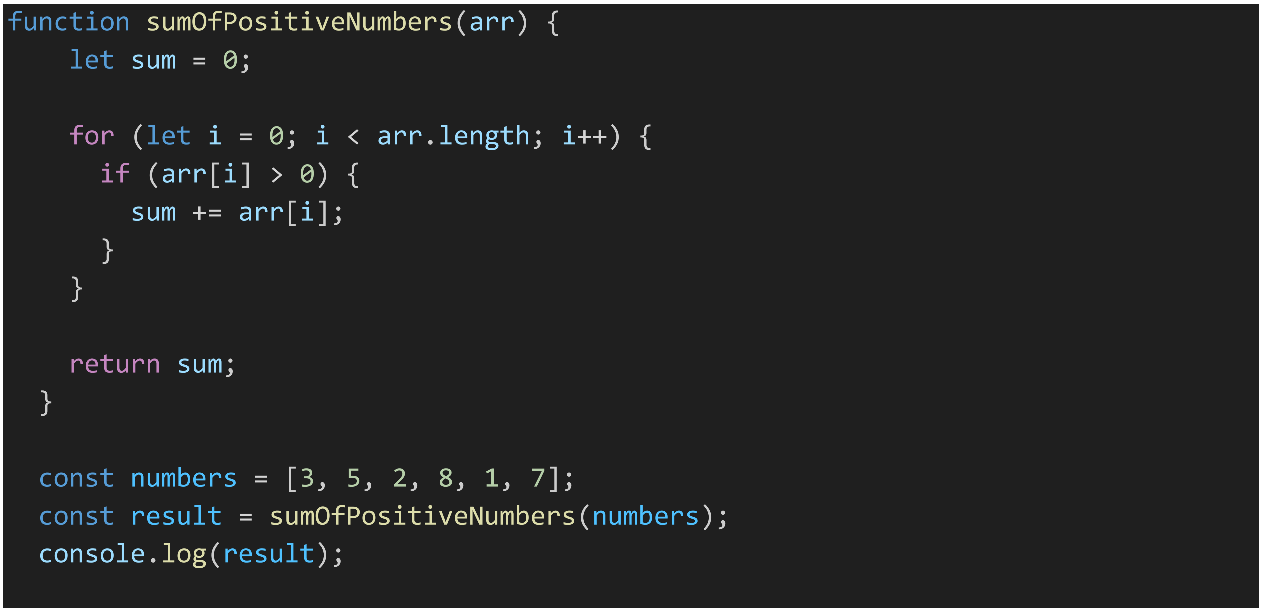
```
    return str === reversedStr;
  }

  console.log(isPalindrome("level"));
  console.log(isPalindrome("hello"));
```

OUTPUT:



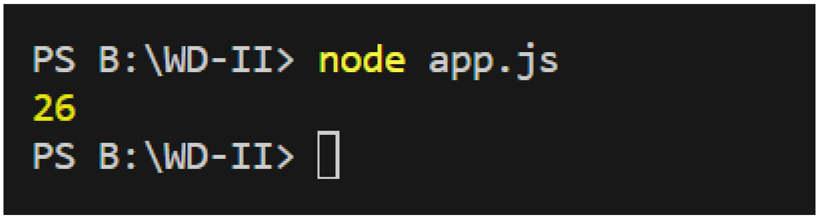
```
PS B:\WD-II> node app.js
true
false
PS B:\WD-II> 
```

5. Question: Find the sum of all positive numbers in an array.



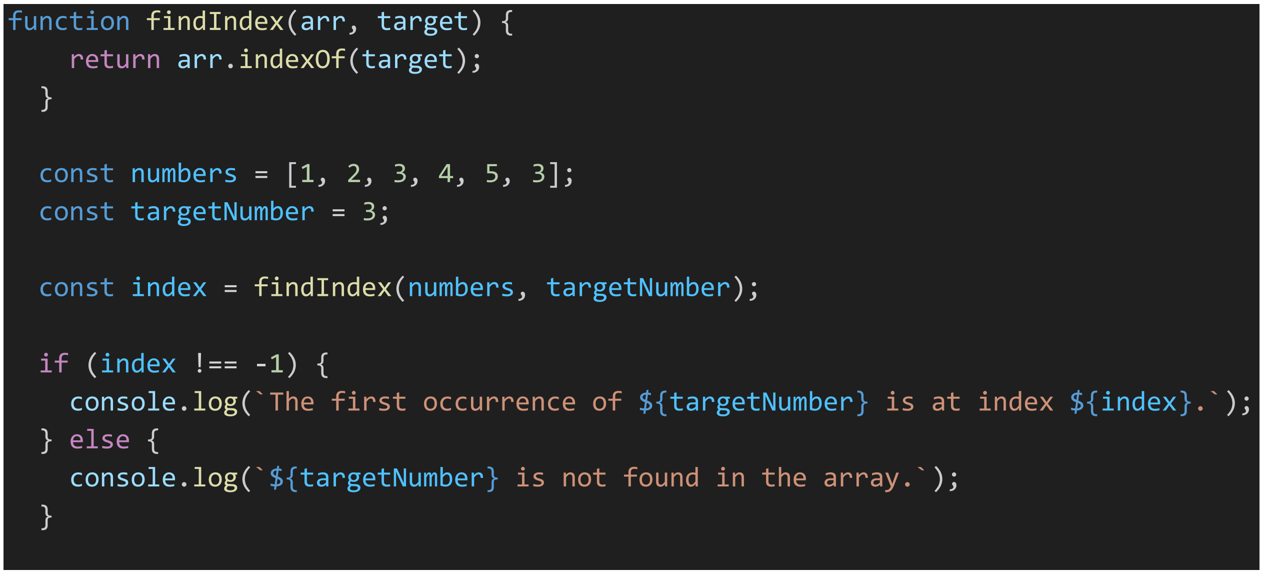
```
function sumOfPositiveNumbers(arr) {
    let sum = 0;

    for (let i = 0; i < arr.length; i++) {
      if (arr[i] > 0) {
        sum += arr[i];
      }
    }

    return sum;
  }

  const numbers = [3, 5, 2, 8, 1, 7];
  const result = sumOfPositiveNumbers(numbers);
  console.log(result);
```

OUTPUT:



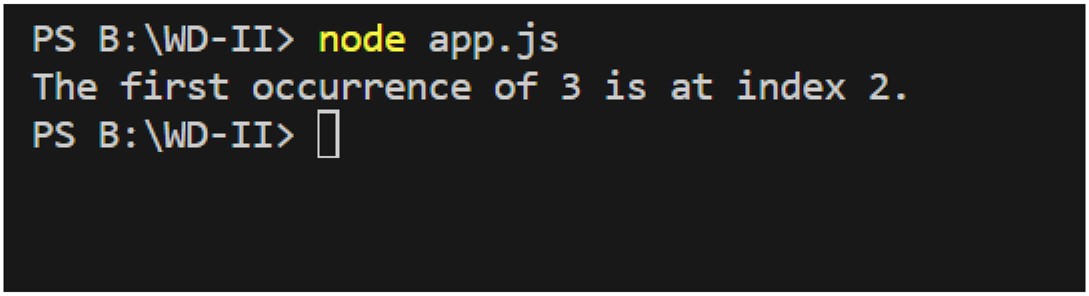
```
PS B:\WD-II> node app.js
26
PS B:\WD-II> 
```

6. Question: Find the index of the first occurrence of a specific element in an array.



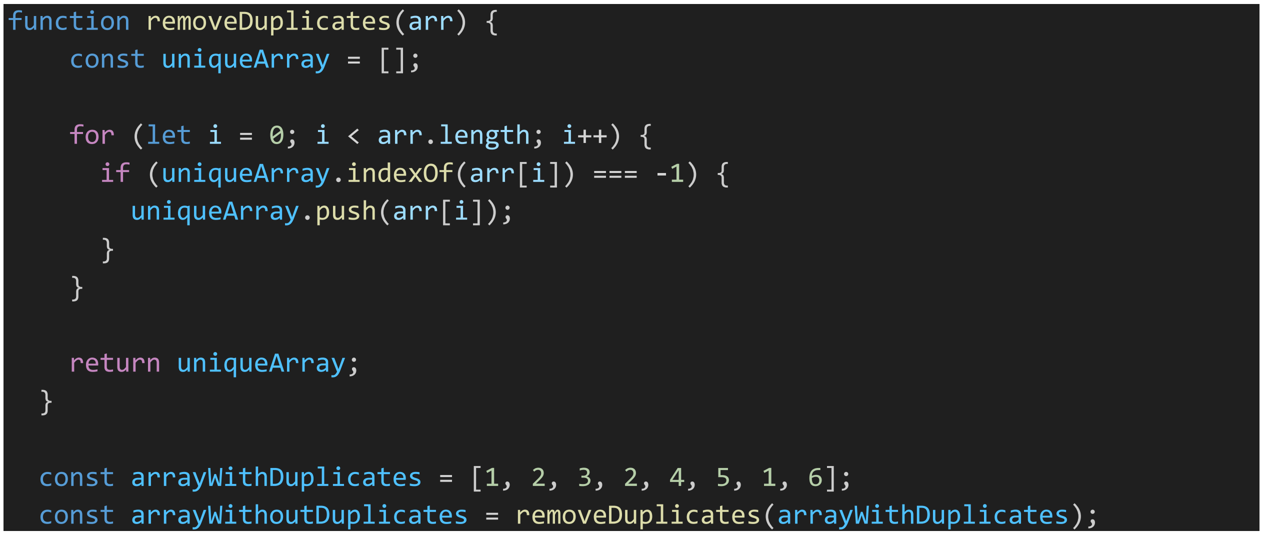
```
function findIndex(arr, target) {
    return arr.indexOf(target);
  }

  const numbers = [1, 2, 3, 4, 5, 3];
  const targetNumber = 3;

  const index = findIndex(numbers, targetNumber);

  if (index !== -1) {
    console.log(`The first occurrence of ${targetNumber} is at index ${index}.`);
  } else {
    console.log(`${targetNumber} is not found in the array.`);
  }
```

OUTPUT:

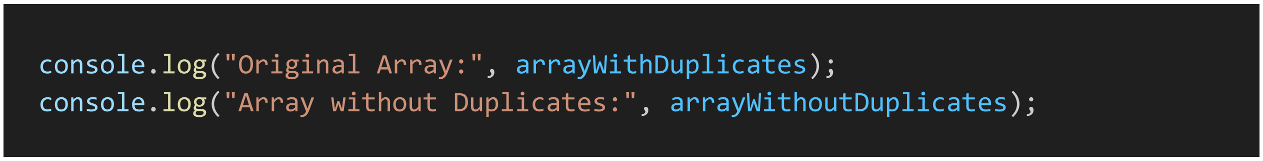


```
PS B:\WD-II> node app.js
The first occurrence of 3 is at index 2.
PS B:\WD-II> 
```

7. Question: Remove all duplicates from an array without built-in methods.

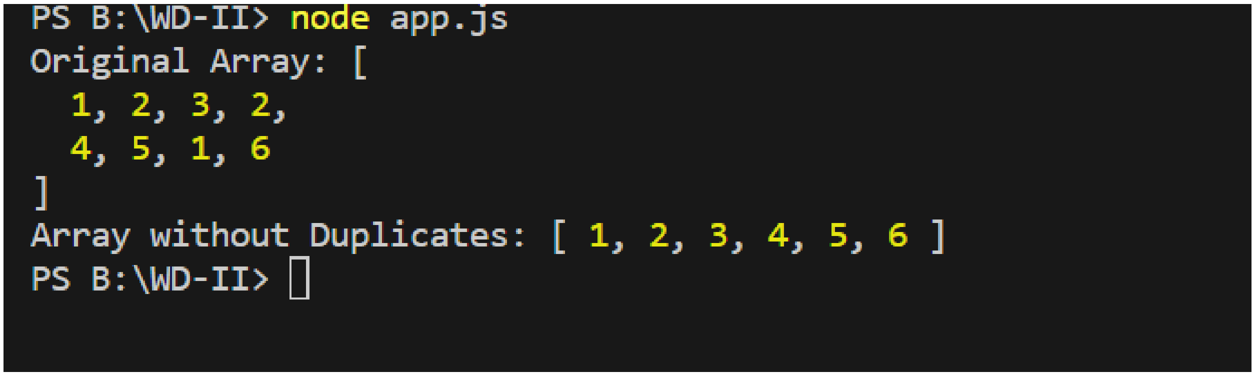


```
function removeDuplicates(arr) {
    const uniqueArray = [];

    for (let i = 0; i < arr.length; i++) {
      if (uniqueArray.indexOf(arr[i]) === -1) {
        uniqueArray.push(arr[i]);
      }
    }

    return uniqueArray;
  }

  const arrayWithDuplicates = [1, 2, 3, 2, 4, 5, 1, 6];
  const arrayWithoutDuplicates = removeDuplicates(arrayWithDuplicates);
```



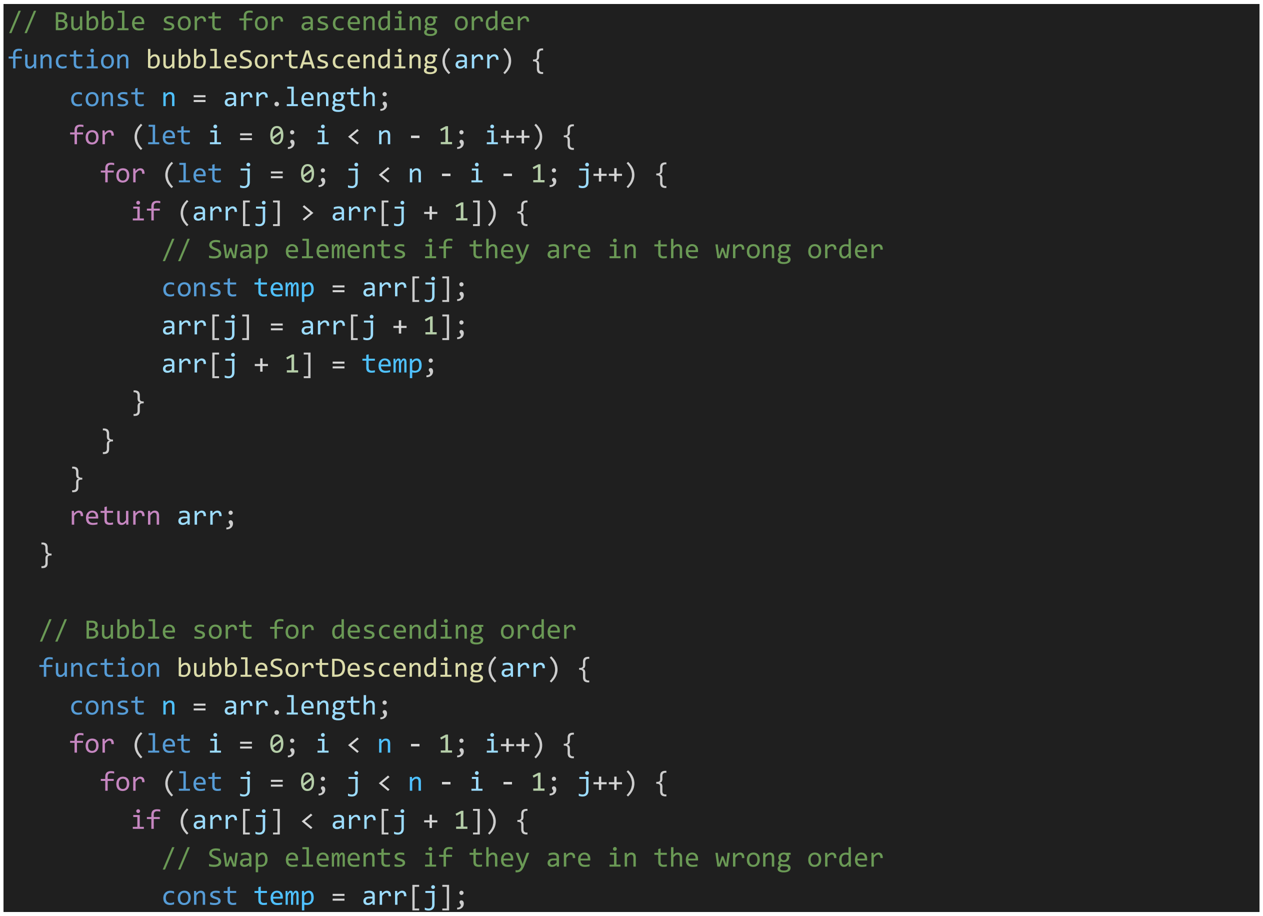
```
console.log("Original Array:", arrayWithDuplicates);
console.log("Array without Duplicates:", arrayWithoutDuplicates);
```

OUTPUT:

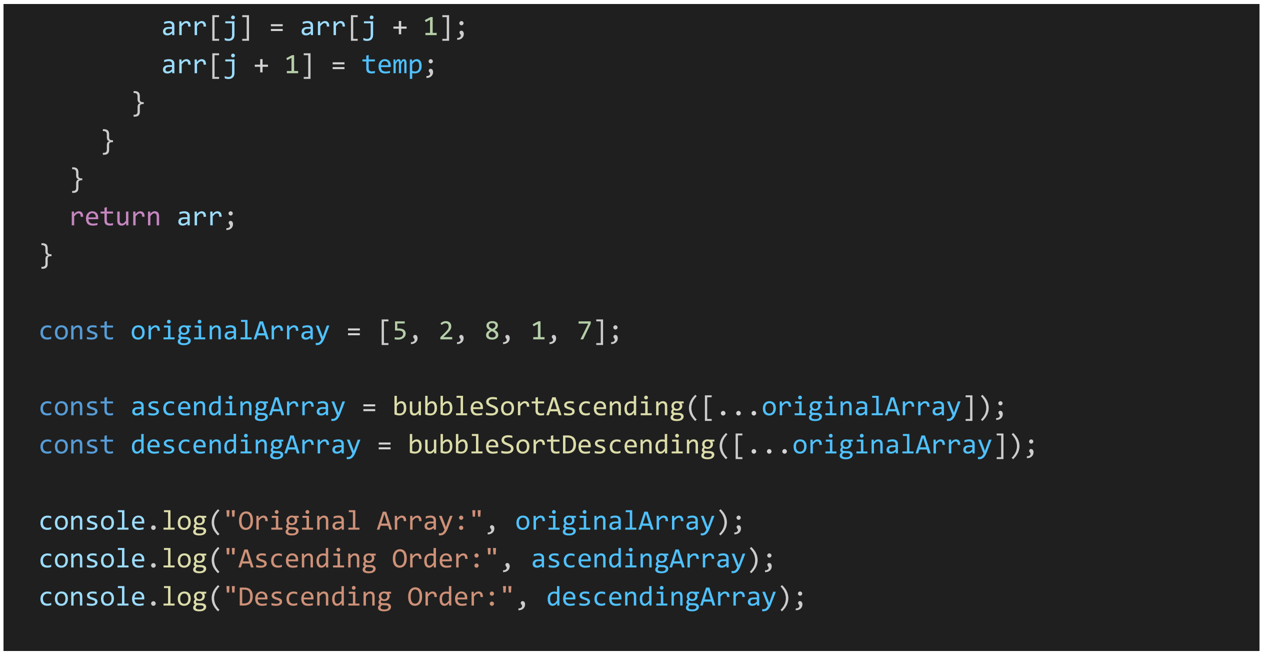


```
PS B:\WD-II> node app.js
Original Array: [
  1, 2, 3, 2,
  4, 5, 1, 6
]
Array without Duplicates: [ 1, 2, 3, 4, 5, 6 ]
PS B:\WD-II> 
```

8. Question: Sort the array in ascending and descending without built-in methods.

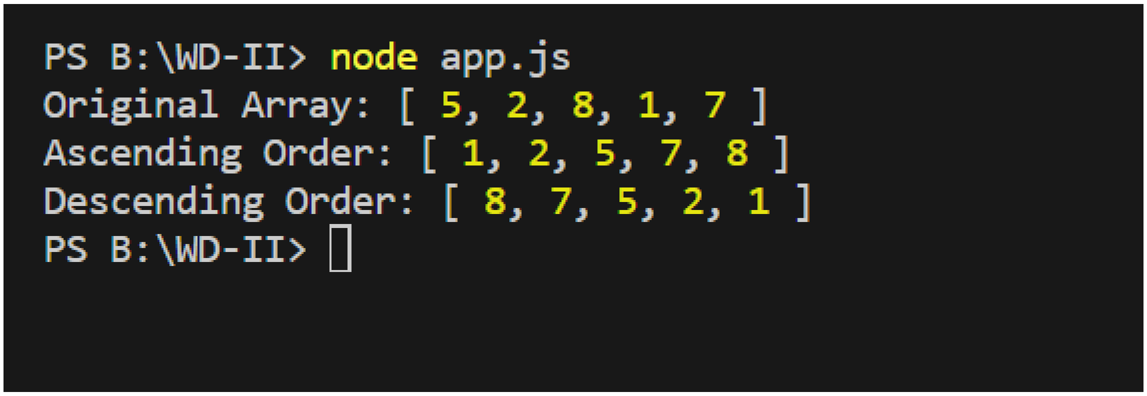


```
// Bubble sort for ascending order
function bubbleSortAscending(arr) {
  const n = arr.length;
  for (let i = 0; i < n - 1; i++) {
   for (let j = 0; j < n - i - 1; j++) {
    if (arr[j] > arr[j + 1]) {
     // Swap elements if they are in the wrong order
     const temp = arr[j];
     arr[j] = arr[j + 1];
     arr[j + 1] = temp;
    }
   }
  }
  return arr;
 }

 // Bubble sort for descending order
 function bubbleSortDescending(arr) {
  const n = arr.length;
  for (let i = 0; i < n - 1; i++) {
   for (let j = 0; j < n - i - 1; j++) {
    if (arr[j] < arr[j + 1]) {
     // Swap elements if they are in the wrong order
     const temp = arr[j];
```



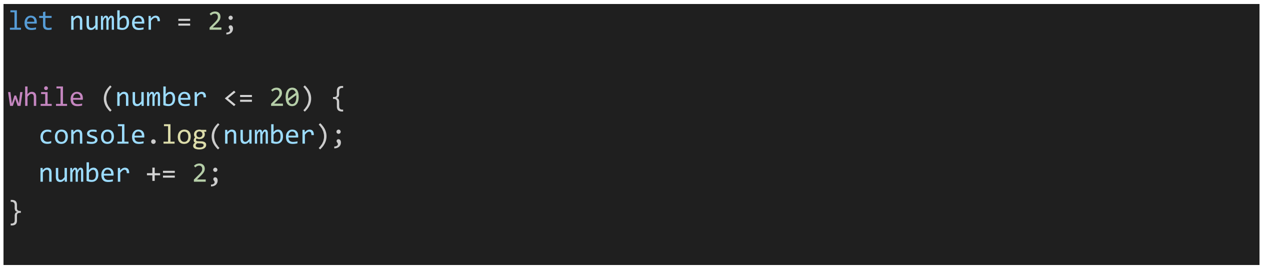
```
        arr[j] = arr[j + 1];
        arr[j + 1] = temp;
      }
    }
  }
  return arr;
}

const originalArray = [5, 2, 8, 1, 7];

const ascendingArray = bubbleSortAscending([...originalArray]);
const descendingArray = bubbleSortDescending([...originalArray]);

console.log("Original Array:", originalArray);
console.log("Ascending Order:", ascendingArray);
console.log("Descending Order:", descendingArray);
```

OUTPUT:



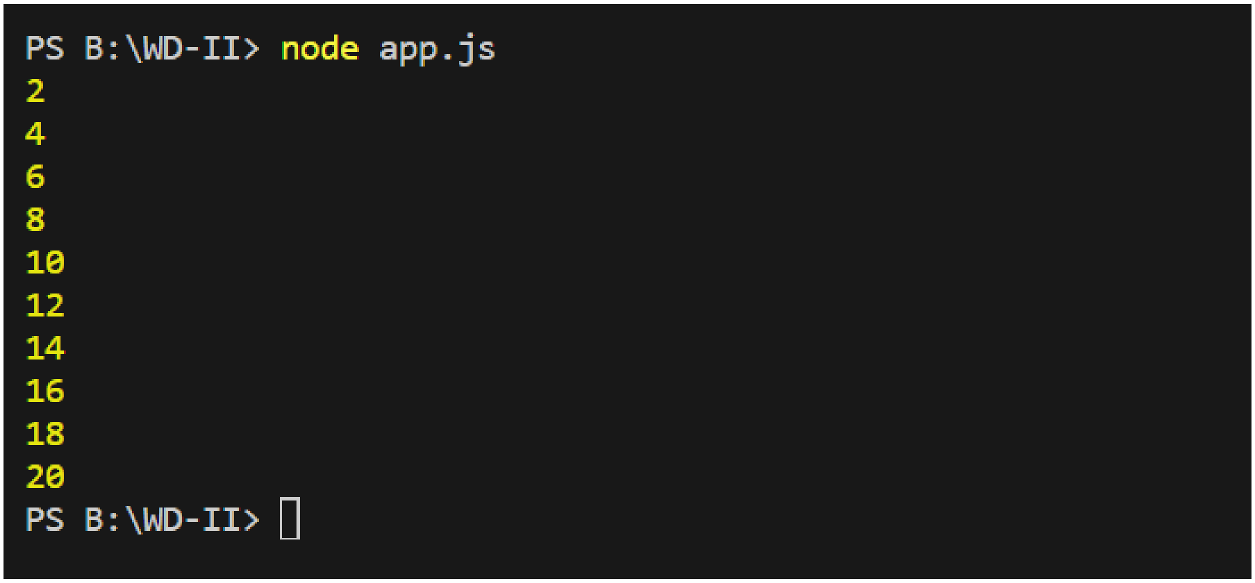
```
PS B:\WD-II> node app.js
Original Array: [ 5, 2, 8, 1, 7 ]
Ascending Order: [ 1, 2, 5, 7, 8 ]
Descending Order: [ 8, 7, 5, 2, 1 ]
PS B:\WD-II> 
```

9. Question: Print all even numbers between 1 and 20 using a while loop.



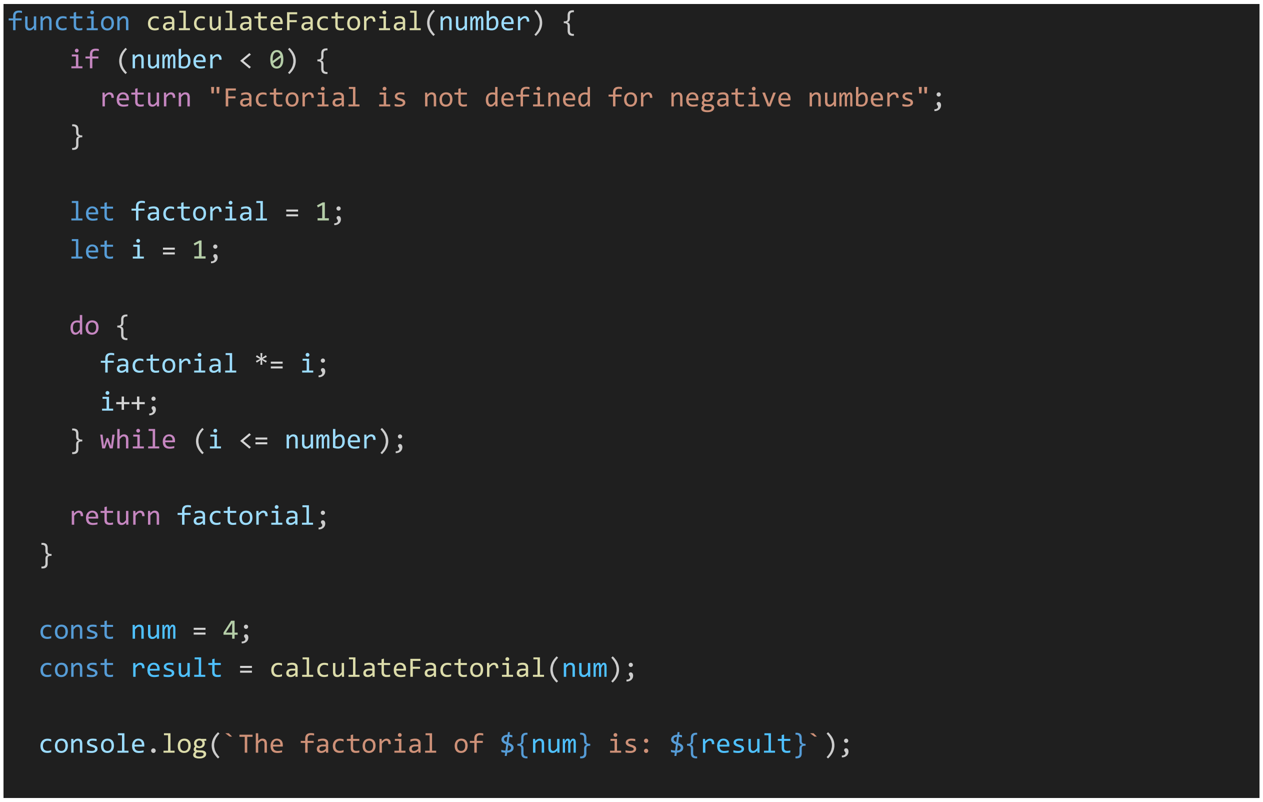
```
let number = 2;

while (number <= 20) {
  console.log(number);
  number += 2;
}
```

OUTPUT:



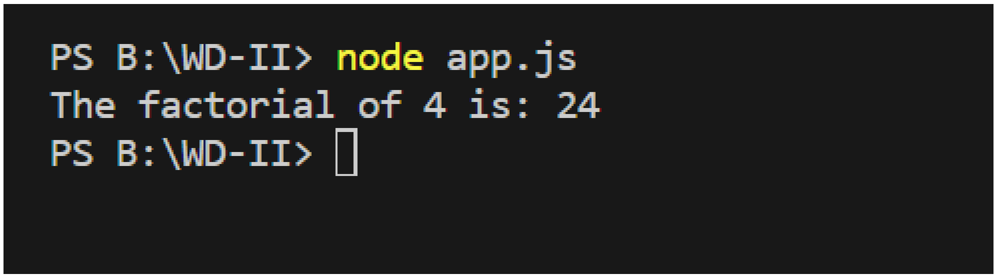
```
PS B:\WD-II> node app.js
2
4
6
8
10
12
14
16
18
20
PS B:\WD-II> 
```

10. Question: Calculate the factorial of a number using a do-while loop.



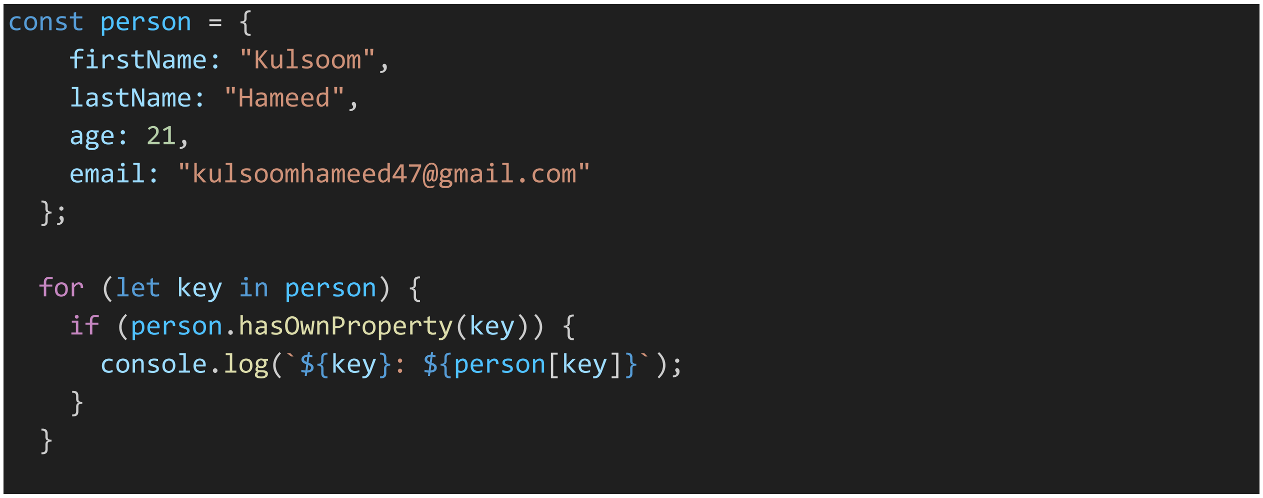
```javascript
function calculateFactorial(number) {
    if (number < 0) {
      return "Factorial is not defined for negative numbers";
    }

    let factorial = 1;
    let i = 1;

    do {
      factorial *= i;
      i++;
    } while (i <= number);

    return factorial;
  }

  const num = 4;
  const result = calculateFactorial(num);

  console.log(`The factorial of ${num} is: ${result}`);
```

OUTPUT:



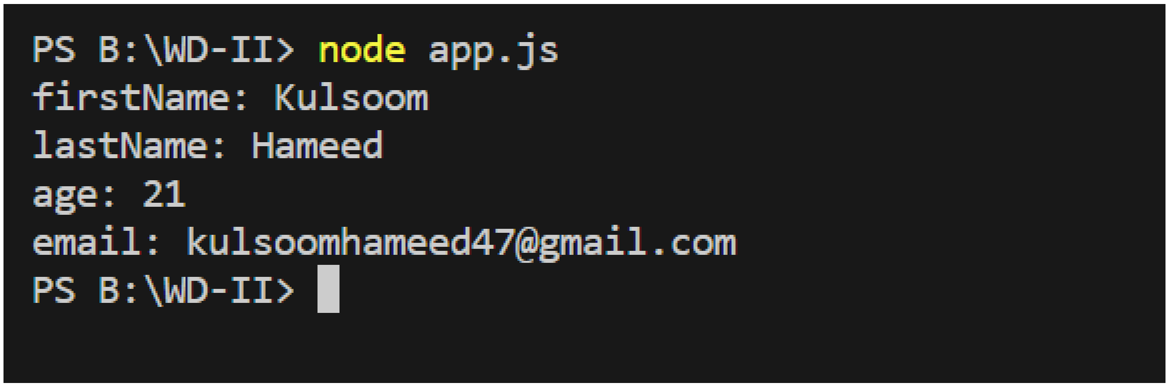
```
PS B:\WD-II> node app.js
The factorial of 4 is: 24
PS B:\WD-II> 
```

11. Question: Iterate through the properties of an object using a for-in loop.



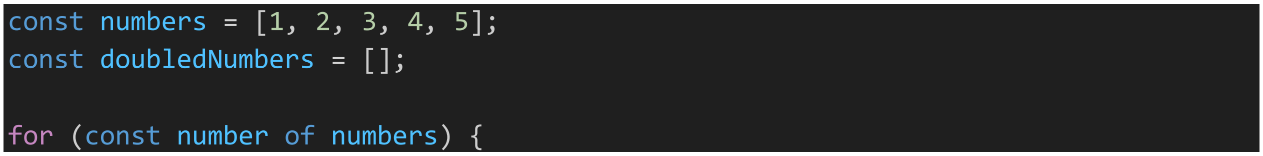
```
const person = {
    firstName: "Kulsoom",
    lastName: "Hameed",
    age: 21,
    email: "kulsoomhameed47@gmail.com"
  };

  for (let key in person) {
    if (person.hasOwnProperty(key)) {
      console.log(`${key}: ${person[key]}`);
    }
  }
```

OUTPUT:

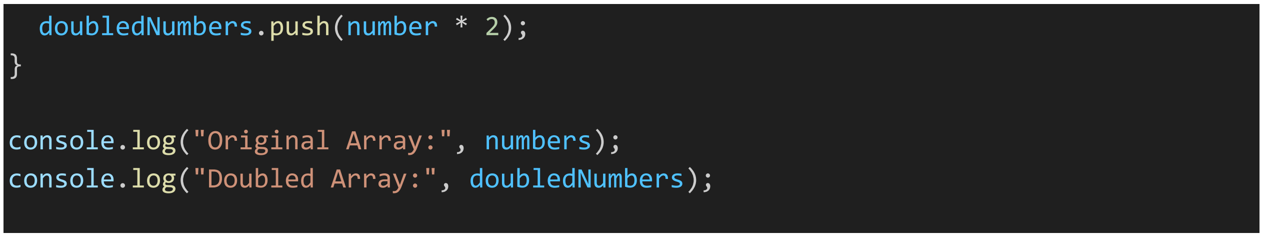


```
PS B:\WD-II> node app.js
firstName: Kulsoom
lastName: Hameed
age: 21
email: kulsoomhameed47@gmail.com
PS B:\WD-II> 
```

12. Question: Loop through an array using a for-of loop and double each element.

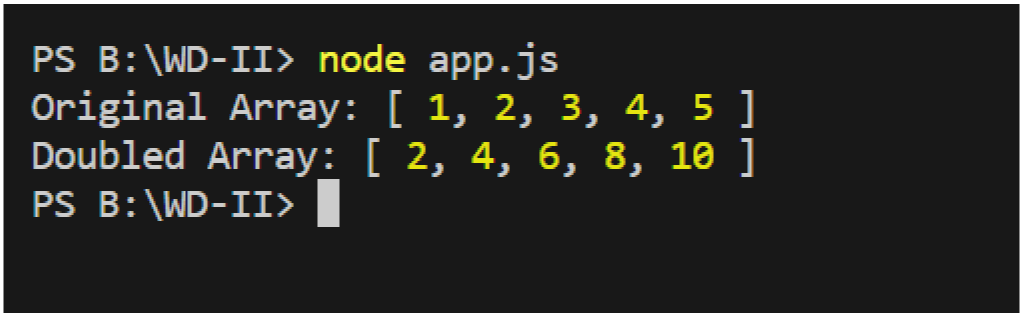


```
const numbers = [1, 2, 3, 4, 5];
const doubledNumbers = [];

for (const number of numbers) {
```



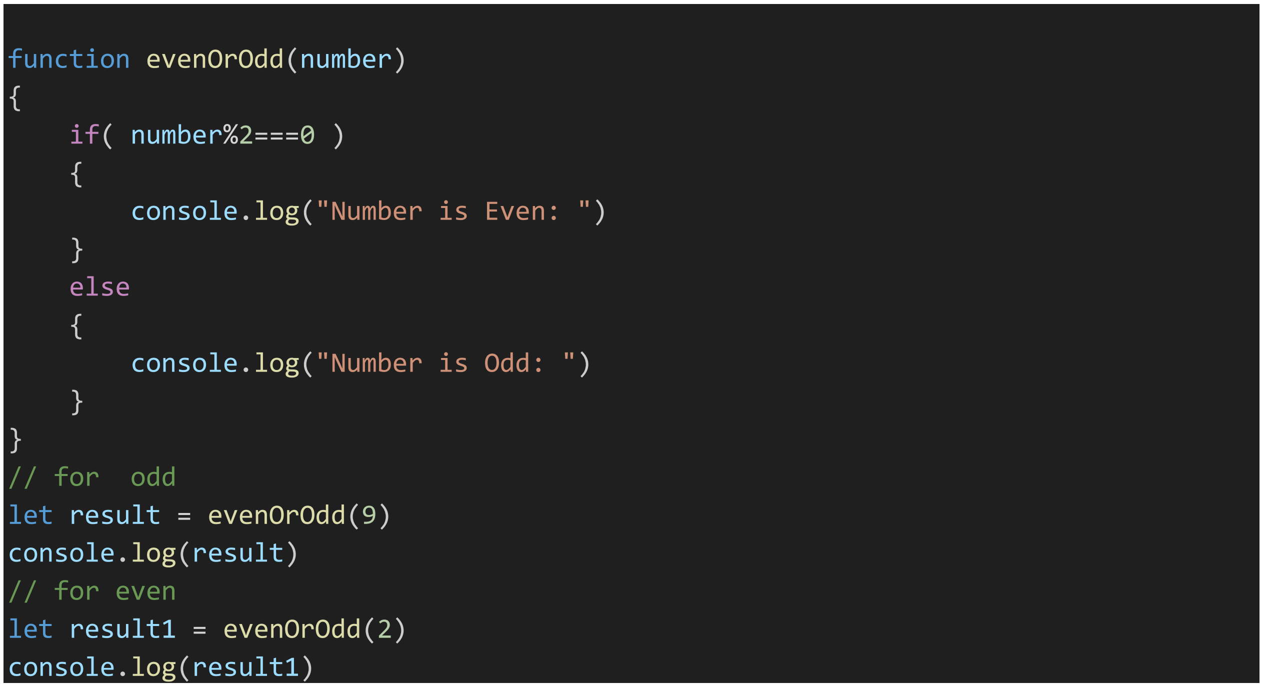
```
  doubledNumbers.push(number * 2);
}

console.log("Original Array:", numbers);
console.log("Doubled Array:", doubledNumbers);
```

OUTPUT:



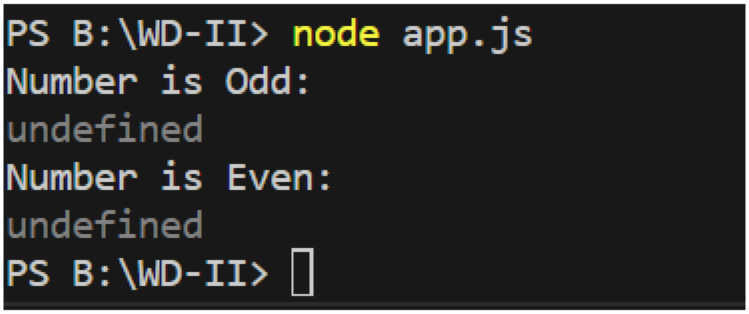
```
PS B:\WD-II> node app.js
Original Array: [ 1, 2, 3, 4, 5 ]
Doubled Array: [ 2, 4, 6, 8, 10 ]
PS B:\WD-II> 
```

13. Question: Check if a number is even or odd and return a corresponding message.



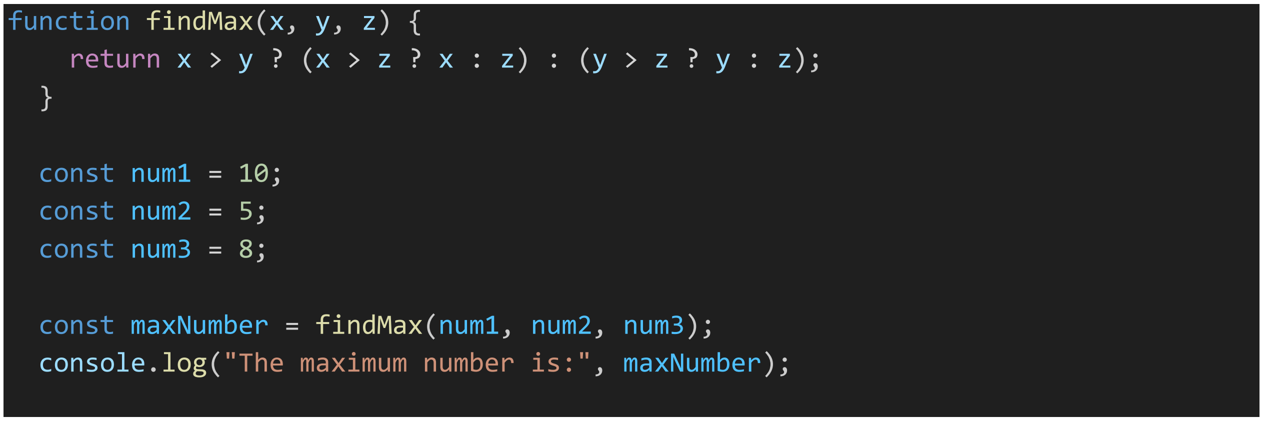
```
function evenOrOdd(number)
{
    if( number%2===0 )
    {
        console.log("Number is Even: ")
    }
    else
    {
        console.log("Number is Odd: ")
    }
}
// for  odd
let result = evenOrOdd(9)
console.log(result)
// for even
let result1 = evenOrOdd(2)
console.log(result1)
```

OUTPUT:



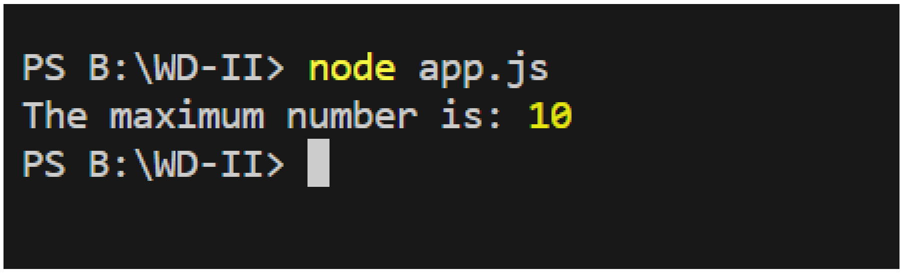
```
PS B:\WD-II> node app.js
Number is Odd:
undefined
Number is Even:
undefined
PS B:\WD-II> 
```

14. Question: Find the maximum of three numbers using nested ternary operators.



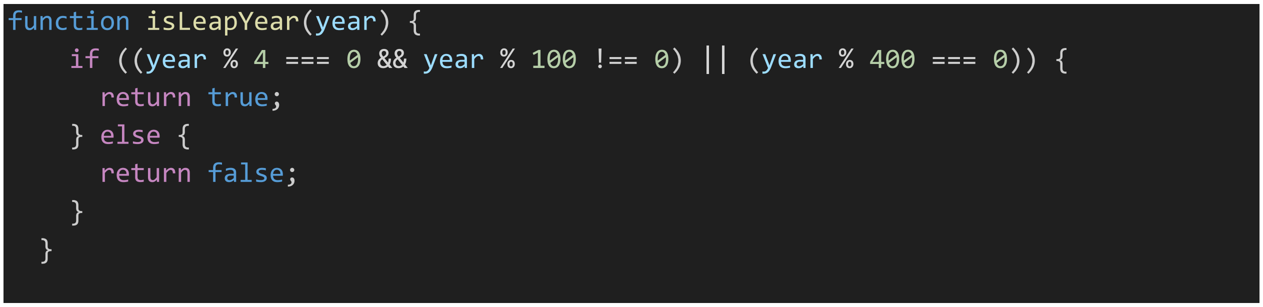
```
function findMax(x, y, z) {
    return x > y ? (x > z ? x : z) : (y > z ? y : z);
  }

  const num1 = 10;
  const num2 = 5;
  const num3 = 8;

  const maxNumber = findMax(num1, num2, num3);
  console.log("The maximum number is:", maxNumber);
```

OUTPUT:



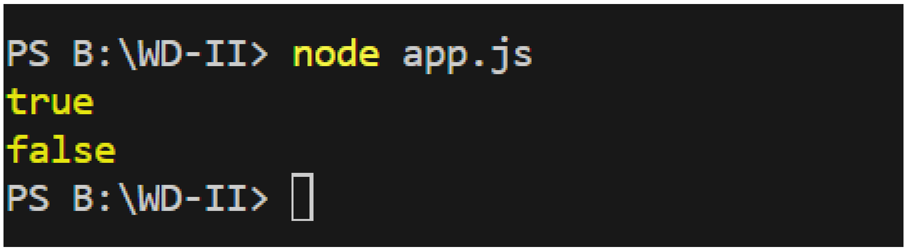
```
PS B:\WD-II> node app.js
The maximum number is: 10
PS B:\WD-II> 
```

15. Question: Determine if a year is a leap year or not.



```
function isLeapYear(year) {
    if ((year % 4 === 0 && year % 100 !== 0) || (year % 400 === 0)) {
      return true;
    } else {
      return false;
    }
  }
  console.log(isLeapYear(2020));
  console.log(isLeapYear(2022));
```

OUTPUT:



```
PS B:\WD-II> node app.js
true
false
PS B:\WD-II> 
```