# **Прикладное программирование**

Практика 24

JavaScript: анимация на web-странице

Для каждой программы, состоящей из нескольких файлов,

создавайте отдельную папку проекта, например project24-1,

project24-2 и т.д., где "24" - номер практического

занятия, 1, 2 и т.д. - номер задания.

Каждую программу, состоящую из одного файла, сохранять

в отдельном файле для каждого упражнения,

например prog24-1, prog24-2 и т.д., где "24" - номер практического

занятия, 1, 2 и т.д. - номер задания.

1. Оформить решение упражнения в виде сниппета в он-лайн HTML-редакоторе:

codepen.io, или jsfiddle.net и др.

В отчет записать ссылку на сниппет, например,

https://codepen.io/Subbota/pen/MRvOOG

Также в отчет поместить снимок страницы с выполненными примерами.

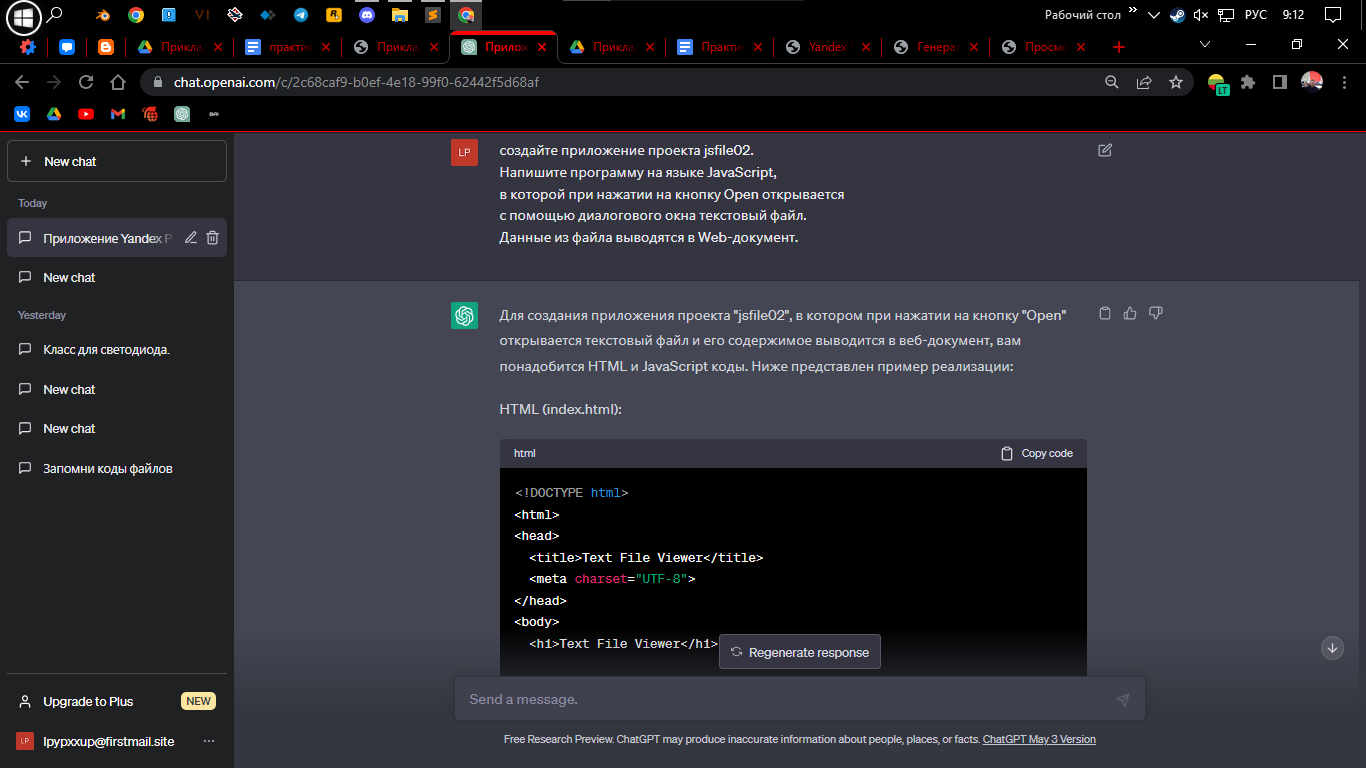
По примеру 33 из лекции создайте приложение проекта jsfile02.

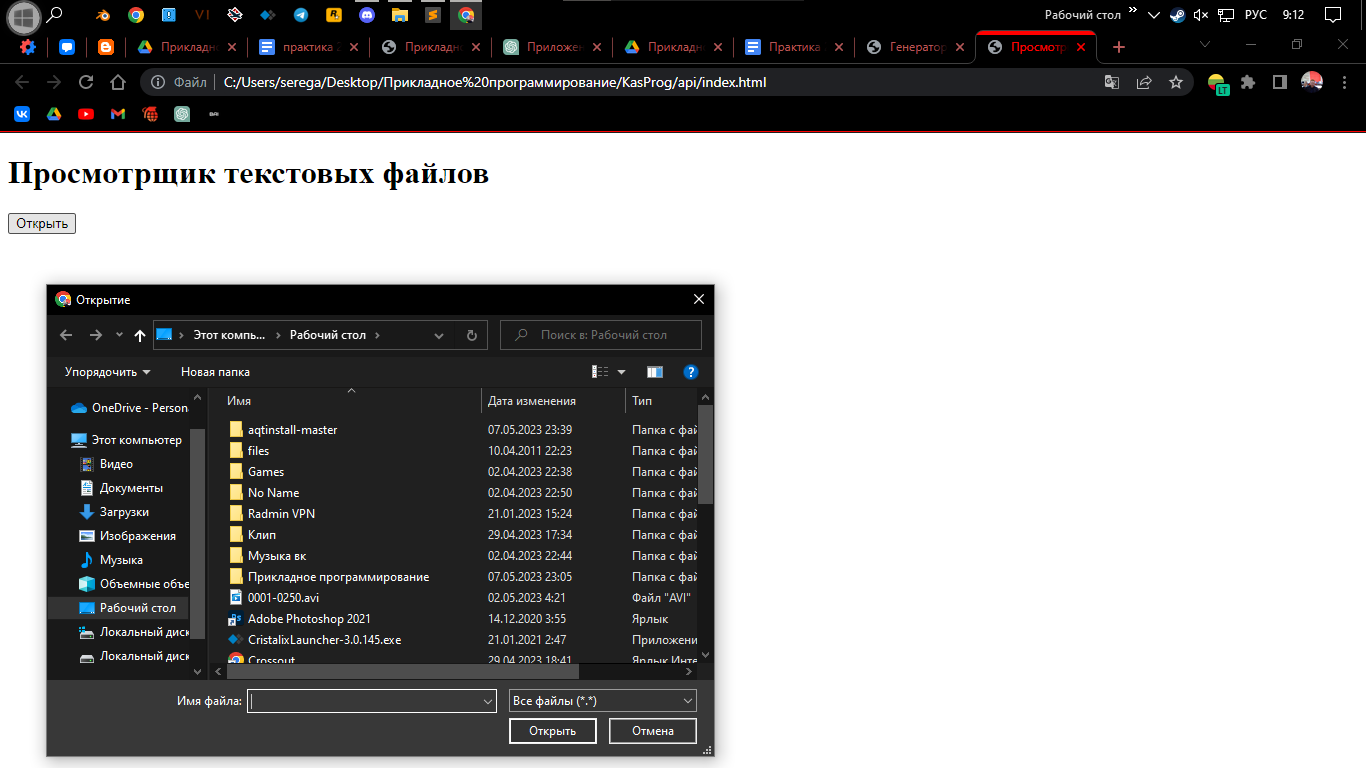
Напишите программу на языке JavaScript,

в которой при нажатии на кнопку Open открывается

с помощью диалогового окна текстовый файл.

Данные из файла выводятся в Web-документ.





<!DOCTYPE html>

<html>

<head>

<title>Text File Viewer</title>

<meta charset="UTF-8">

</head>

<body>

<h1>Text File Viewer</h1>

<button onclick="openFile()">Open</button>

<div id="fileContent"></div>

<script src="app.js"></script>

</body>

</html>

function openFile() {

// Создаем input элемент для выбора файла

var input = document.createElement('input');

input.type = 'file';

// Добавляем обработчик события выбора файла

input.addEventListener('change', function(event) {

var file = event.target.files[0];

// Создаем объект FileReader для чтения содержимого файла

var reader = new FileReader();

// Определяем обработчик события загрузки файла

reader.onload = function(event) {

var content = event.target.result;

// Выводим содержимое файла на страницу

var fileContentElement = document.getElementById('fileContent');

fileContentElement.textContent = content;

};

// Читаем содержимое файла в формате текста

reader.readAsText(file);

});

// Эмулируем нажатие на input элемент для открытия диалогового окна выбора файла

input.click();

}

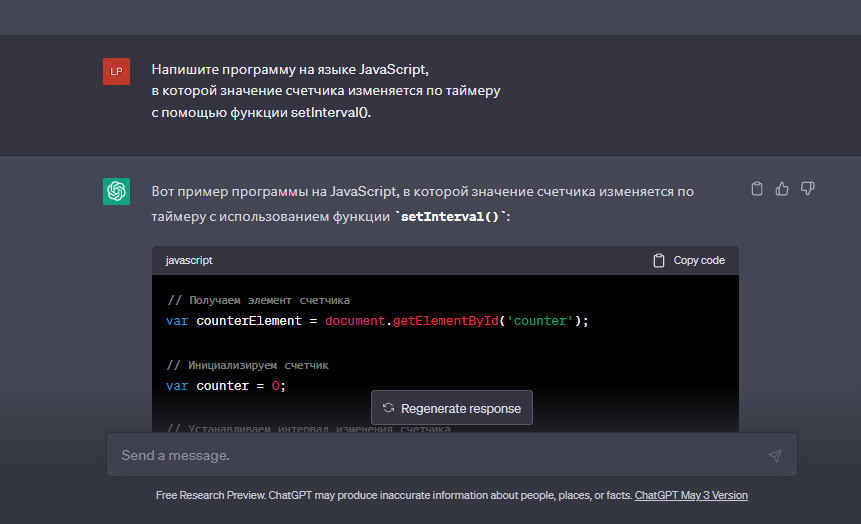
2. На основе примера 34 из лекции создайте проект jstimer02.

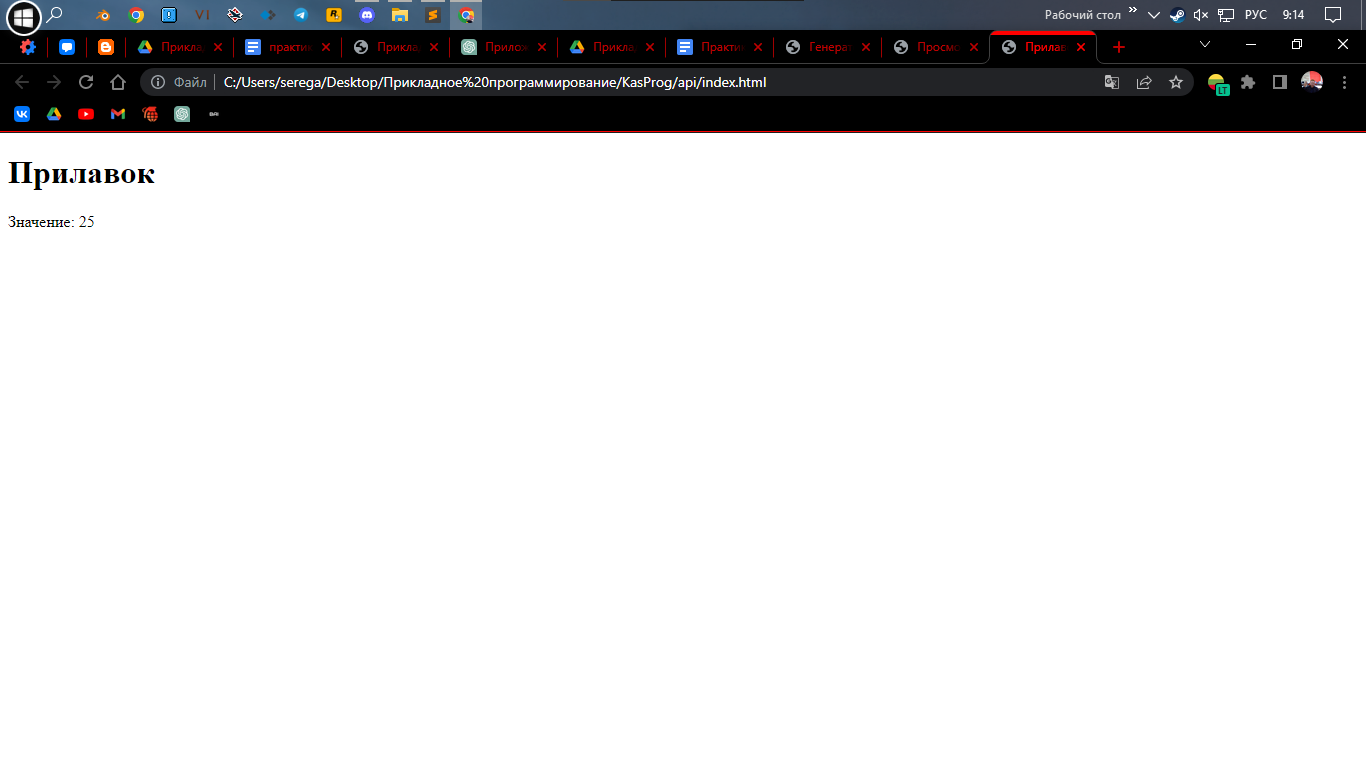
Выполните задание в онлайн средах codepen.io, или jsfiddle.net и др.

Напишите программу на языке JavaScript,

в которой значение счетчика изменяется по таймеру

с помощью функции setInterval().





<!DOCTYPE html>

<html>

<head>

<title>Counter</title>

<meta charset="UTF-8">

</head>

<body>

<h1>Counter</h1>

<p>Value: <span id="counter">0</span></p>

<script src="app.js"></script>

</body>

</html>

// Получаем элемент счетчика

var counterElement = document.getElementById('counter');

// Инициализируем счетчик

var counter = 0;

// Устанавливаем интервал изменения счетчика

var intervalId = setInterval(function() {

// Увеличиваем значение счетчика

counter++;

// Обновляем текст элемента счетчика

counterElement.textContent = counter;

}, 1000); // Изменение счетчика каждую секунду

3. Используйте функцию setInterval() для решения следующей задачи на языке JavaScript.

Выполните задание в онлайн средах codepen.io, или jsfiddle.net и др.

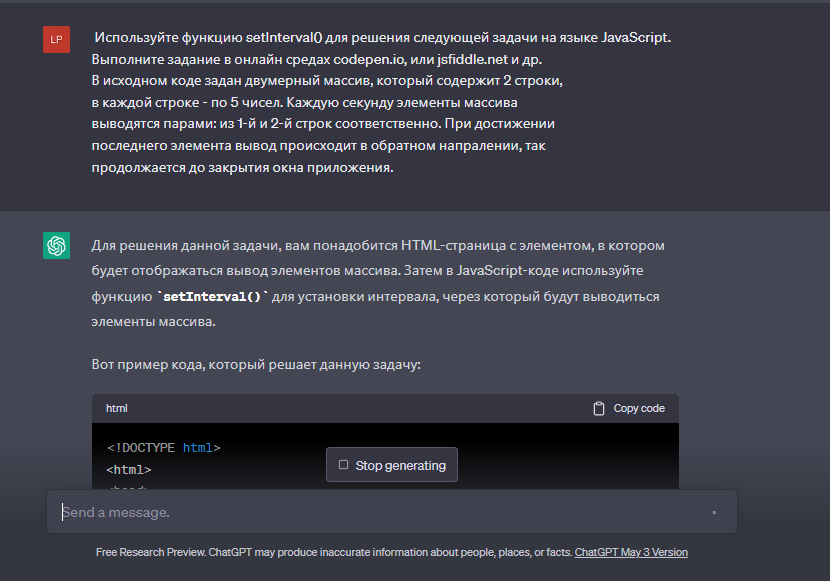
В исходном коде задан двумерный массив, который содержит 2 строки,

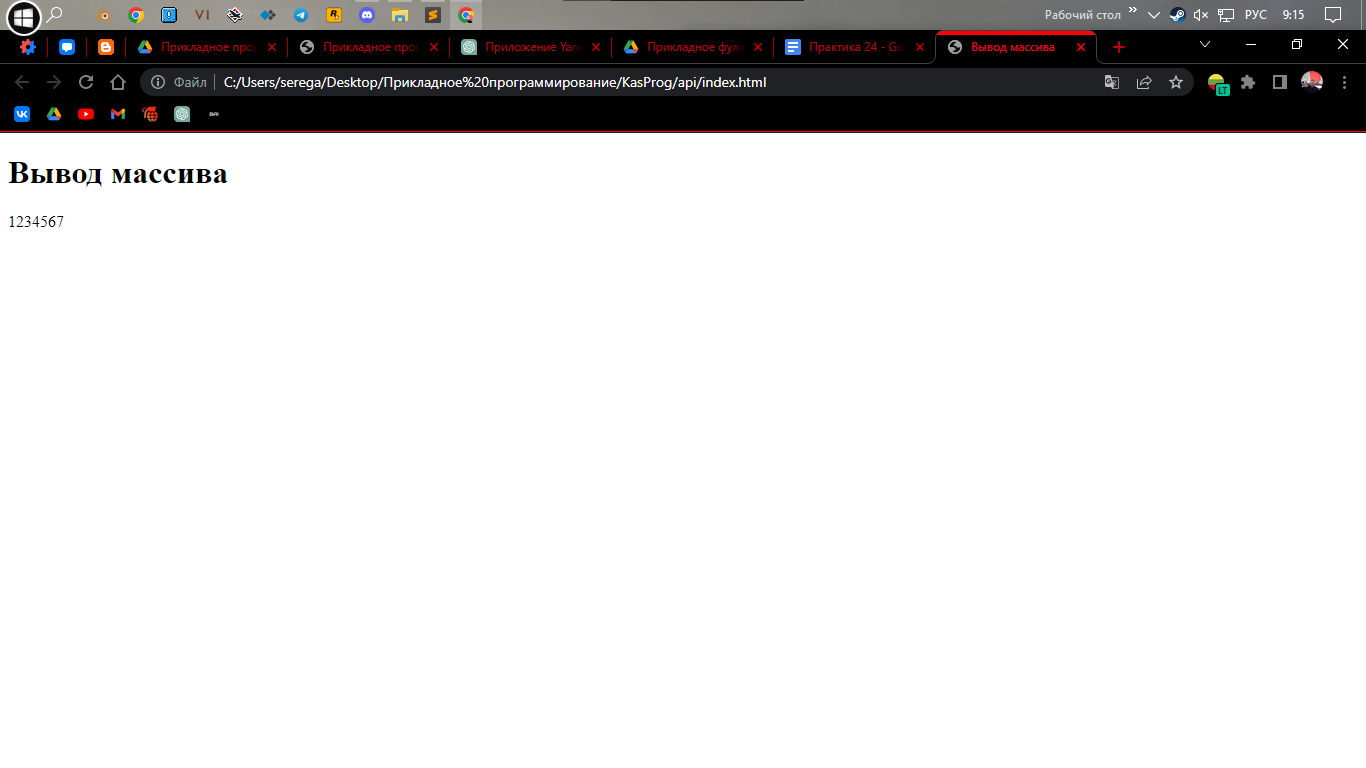
в каждой строке - по 5 чисел. Каждую секунду элементы массива

выводятся парами: из 1-й и 2-й строк соответственно. При достижении

последнего элемента вывод происходит в обратном напралении, так

продолжается до закрытия окна приложения.





<!DOCTYPE html>

<html>

<head>

<title>Array Output</title>

<meta charset="UTF-8">

</head>

<body>

<h1>Array Output</h1>

<div id="output"></div>

<script src="app.js"></script>

</body>

</html>

// Исходный двумерный массив

var array = [

[1, 2, 3, 4, 5],

[6, 7, 8, 9, 10]

];

// Получаем элемент вывода

var outputElement = document.getElementById('output');

// Устанавливаем начальные значения индексов

var row = 0;

var col = 0;

// Устанавливаем интервал вывода элементов массива

var intervalId = setInterval(function() {

// Выводим текущий элемент

outputElement.textContent += array[row][col] + ' ';

// Увеличиваем индекс столбца

col++;

// Если достигли конца строки, переходим на следующую строку

if (col >= array[row].length) {

row++;

col = 0;

// Если достигли конца массива, останавливаем интервал

if (row >= array.length) {

clearInterval(intervalId);

}

}

}, 1000); // Интервал 1 секунда

4. В папке с проектом находятся десять текстовых файлов: 01.txt,

02.txt, ..., 10.txt. В каждом файле находится строка из 5 чисел,

разделенных запятыми. Каждую секунду сначала выводятся все данные

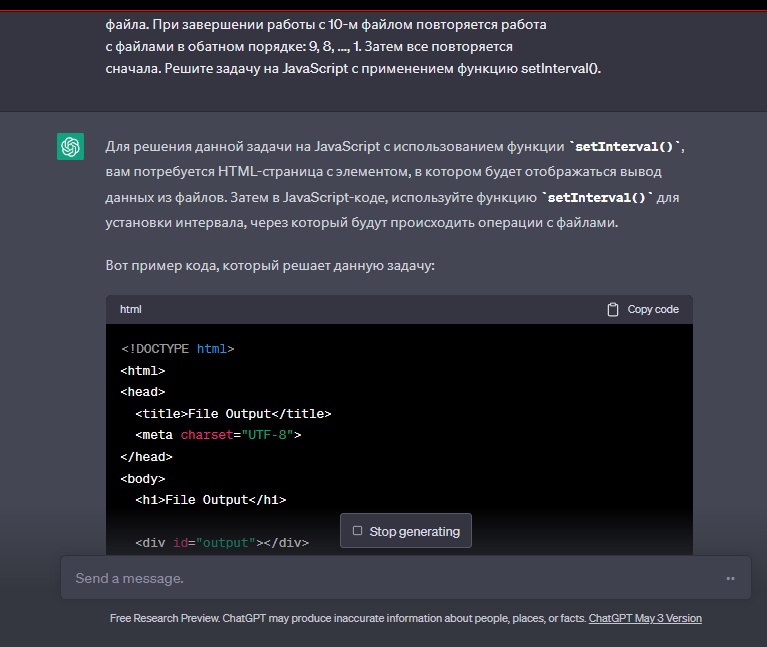
из 1-го файла, затем по одному числу. То же повторяется для каждого

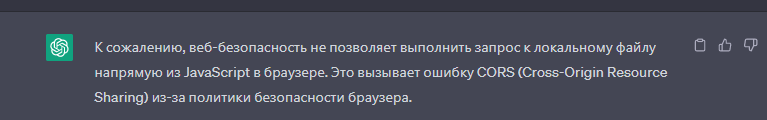
файла. При завершении работы с 10-м файлом повторяется работа

с файлами в обатном порядке: 9, 8, ..., 1. Затем все повторяется

сначала. Решите задачу на JavaScript с применением функцию setInterval().

Выполните задание в онлайн средах codepen.io, или jsfiddle.net и др.



<!DOCTYPE html>

<html>

<head>

<meta charset="UTF-8">

<title>File Reader</title>

</head>

<body>

<h1>File Reader</h1>

<div id="output"></div>

<script>

const files = [

'01.txt', '02.txt', '03.txt', '04.txt', '05.txt',

'06.txt', '07.txt', '08.txt', '09.txt', '10.txt'

];

let currentIndex = 0;

let increment = true;

function readAndPrintFile() {

const fileName = files[currentIndex];

fetch(fileName)

.then(response => response.text())

.then(data => {

const numbers = data.trim().split(',');

const fileOutput = document.createElement('div');

fileOutput.textContent = `File ${fileName}: ${numbers.join(', ')}`;

document.getElementById('output').appendChild(fileOutput);

if (increment) {

numbers.forEach((number, index) => {

setTimeout(() => {

const numberOutput = document.createElement('div');

numberOutput.textContent = `Number from ${fileName}: ${number}`;

document.getElementById('output').appendChild(numberOutput);

}, (index + 1) \* 1000);

});

} else {

for (let i = numbers.length - 1; i >= 0; i--) {

setTimeout(() => {

const numberOutput = document.createElement('div');

numberOutput.textContent = `Number from ${fileName}: ${numbers[i]}`;

document.getElementById('output').appendChild(numberOutput);

}, (numbers.length - i) \* 1000);

}

}

});

currentIndex = increment ? (currentIndex + 1) % files.length : (currentIndex - 1 + files.length) % files.length;

if (currentIndex === 0) {

increment = !increment;

}

}

setInterval(readAndPrintFile, 10000);

</script>

</body>

</html>

const fs = require('fs');

const files = [

'01.txt', '02.txt', '03.txt', '04.txt', '05.txt',

'06.txt', '07.txt', '08.txt', '09.txt', '10.txt'

];

let currentIndex = 0;

let increment = true;

function readAndPrintFile() {

const fileName = files[currentIndex];

const filePath = `./${fileName}`;

fs.readFile(filePath, 'utf8', (err, data) => {

if (err) {

console.error(`Error reading file ${fileName}:`, err);

return;

}

const numbers = data.trim().split(',');

console.log(`File ${fileName}:`, numbers);

if (increment) {

numbers.forEach((number, index) => {

setTimeout(() => {

console.log(`Number from ${fileName}:`, number);

}, (index + 1) \* 1000);

});

} else {

for (let i = numbers.length - 1; i >= 0; i--) {

setTimeout(() => {

console.log(`Number from ${fileName}:`, numbers[i]);

}, (numbers.length - i) \* 1000);

}

}

});

currentIndex = increment ? (currentIndex + 1) % files.length : (currentIndex - 1 + files.length) % files.length;

if (currentIndex === 0) {

increment = !increment;

}

}

setInterval(readAndPrintFile, 10000);