**Name: Tejas M (24TMPMC99)**

**Week-7 WEB Lab Programs**

1. **Use of const,let and template strings**

<!DOCTYPE html>

<html lang="en">

<head>

<title>Template Strings Example</title>

</head>

<body>

<h2>Template Strings in ES6</h2>

<script>

const name = "Tejas";

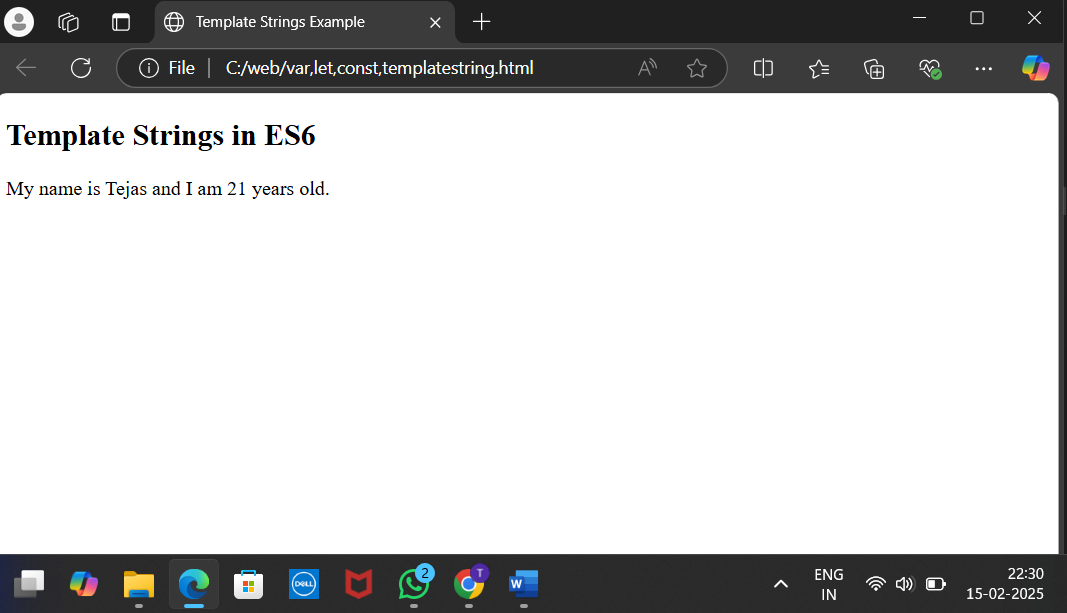
let age = 21;

document.write(`My name is ${name} and I am ${age} years old.`);

</script>

</body>

</html>



1. **Function Scope**

<html>

<head>

    <title>Scope Example</title>

</head>

<body>

    <h2>Function Scope in ES6</h2>

    <script>

        function scopeExample() {

            if (true) {

                var x = 10;

                let y = 20;

                const z = 30;

                document.write(`Inside block: x=${x}, y=${y}, z=${z}<br>`);

            }

            document.write(`Outside block: x=${x}`);

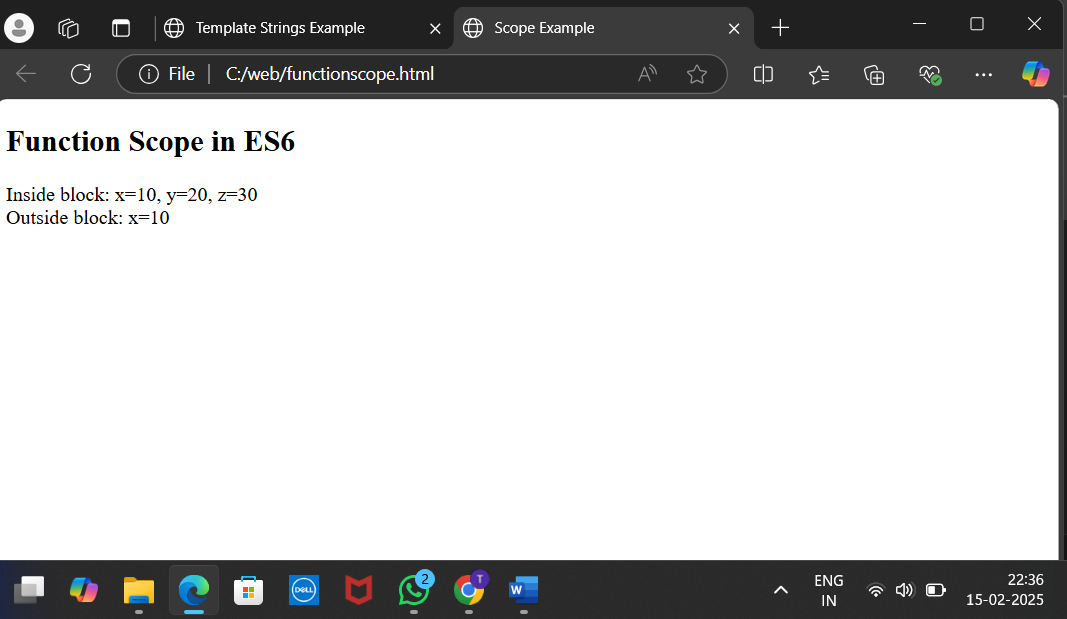
        }

        scopeExample();

    </script>

</body>

</html>



1. Array

<!DOCTYPE html>

<html>

<head><title>Array Operations</title></head>

<body>

    <script>

        let nums = [10, 20, 30, 40, 50];

        nums.push(60);

        document.write(`After push: ${nums} <br>`);

        nums.pop();

        document.write(`After pop: ${nums} <br>`);

        nums.unshift(5);

        document.write(`After unshift: ${nums} <br>`);

        nums.shift();

        document.write(`After shift: ${nums} <br>`);

        let part = nums.slice(1, 3);

        document.write(`Slice (index 1 to 3): ${part} <br>`);

        nums.splice(2, 1, 35);

        document.write(`After splice: ${nums} <br>`);

        let moreNums = [70, 80];

        let combined = nums.concat(moreNums);

        document.write(`After concat: ${combined} <br>`);

        document.write(`Includes 30? ${nums.includes(30)} <br>`);

        let [first, second, ...rest] = nums;

        document.write(`Destructured First: ${first}, Second: ${second} <br>`);

        document.write(`Rest of the array: ${rest} <br>`);

    </script>

</body>

</html>

A screenshot of a computer

AI-generated content may be incorrect.

1. Object Literal Enhancements

<!DOCTYPE html>

<html>

<head><title>Object Literal Enhancements</title></head>

<body>

<script>

let name = "Tejas";

let age = 21;

let course = "MCA";

let student = { name, age, course };

let college = {

name: "BMS College",

location: "Bangalore",

details() {

return `College: ${this.name}, Location: ${this.location}`;

}

};

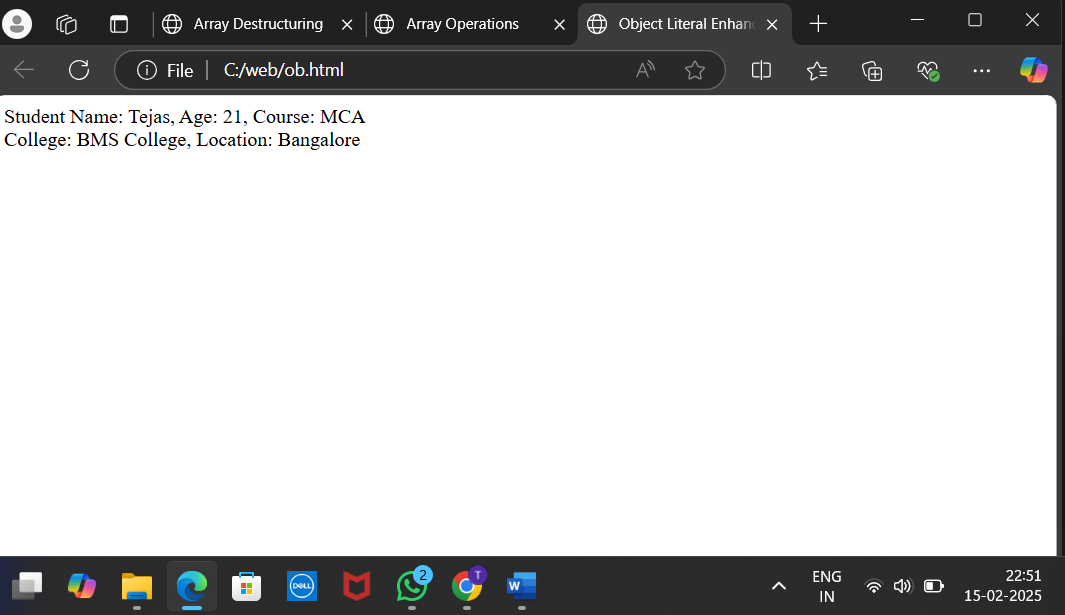
document.write(`Student Name: ${student.name}, Age: ${student.age}, Course: ${student.course} <br>`);

document.write(college.details());

</script>

</body>

</html>



1. Destuvtion example using rest

<!DOCTYPE html>

<html>

<head><title>Destructuring Examples</title></head>

<body>

    <script>

        let student = { name: "Tejas", age: 21, course: "MCA" };

        let { name, age } = student;

        document.write(`Student Name: ${name}, Age: ${age} <br>`);

        let { course = "Not Assigned" } = student;

        document.write(`Course: ${course} <br>`);

        let numbers = [10, 20, 30, 40, 50];

        let [first, second, ...rest] = numbers;

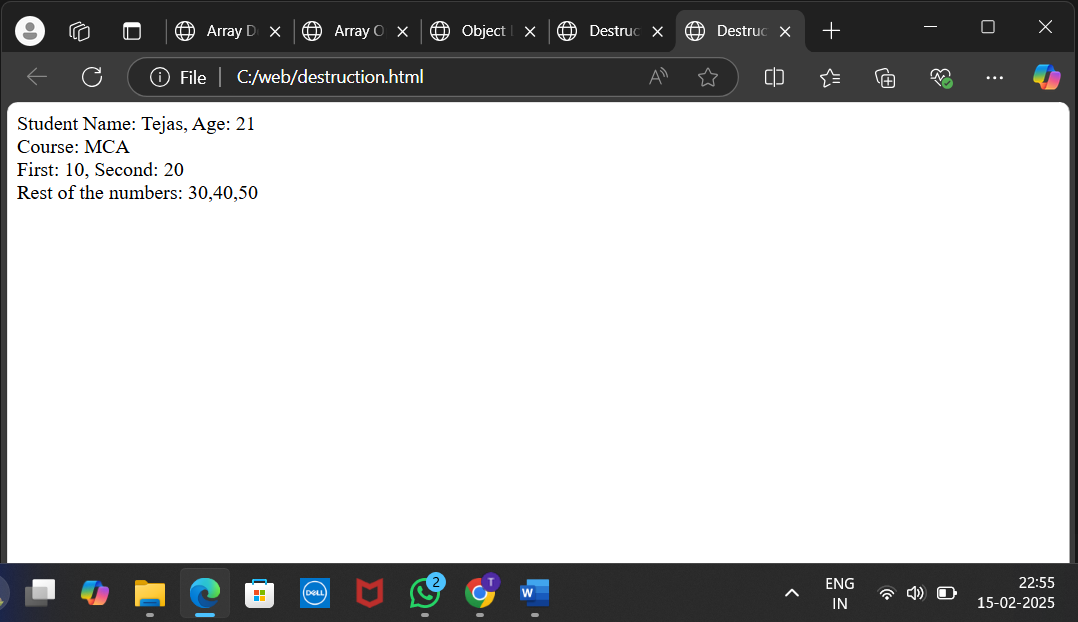
        document.write(`First: ${first}, Second: ${second} <br>`);

        document.write(`Rest of the numbers: ${rest} <br>`);

    </script>

</body>

</html>



1. Understanding var let const and function accessing is ES6

<!DOCTYPE html>

<html>

<head><title>ES6 Var, Let, Const</title></head>

<body>

<script>

var x = 10;

document.write(`Var x: ${x} <br>`);

x = 20;

document.write(`Updated Var x: ${x} <br><br>`);

let y = 30;

document.write(`Let y: ${y} <br>`);

y = 40;

document.write(`Updated Let y: ${y} <br><br>`);

const z = 50;

document.write(`Const z: ${z} <br><br>`);

function testScope() {

var a = 100;

let b = 200;

const c = 300;

document.write(`Inside Function - Var a: ${a}, Let b: ${b}, Const c: ${c} <br>`);

}

testScope();

if (true) {

var d = 400;

let e = 500;

const f = 600;

document.write(`Inside Block - Var d: ${d}, Let e: ${e}, Const f: ${f} <br>`);

}

document.write(`Outside Block - Var d: ${d} <br>`);

</script>

</body>

</html>

