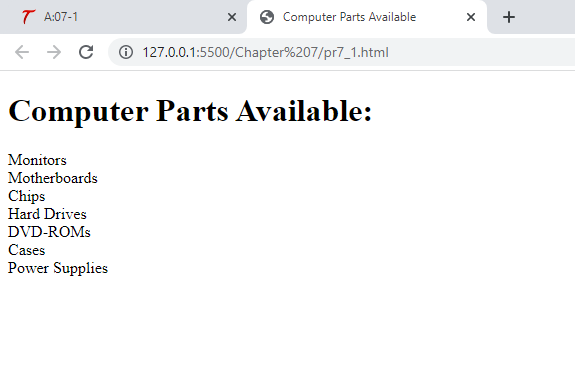
Jesus Nolazco

Chapter 7

**Try This 7-1**



<!DOCTYPE html>

<html>

<head>

    <title>Computer Parts Available</title>

</head>

<body>

    <h1>Computer Parts Available: </h1>

    <script src="prjs7\_1.js"></script>

</body>

</html>

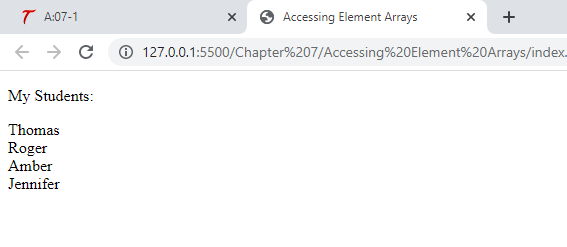
let computer\_parts = ["Monitors", "Motherboards", "Chips", "Hard Drives", "DVD-ROMs", "Cases", "Power Supplies"];

for (let parts = 0; parts < computer\_parts.length; parts++) {

    document.write(computer\_parts[parts] + "<br>");

}

**Page 167 Accessing array elements**

****

<!DOCTYPE html>

<html>

<head>

    <title>Accessing Element Arrays</title>

</head>

<body>

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

</body>

</html>

let students = ["Thomas", "Roger", "Amber", "Jennifer"];

document.write("<p>My Students:</p>");

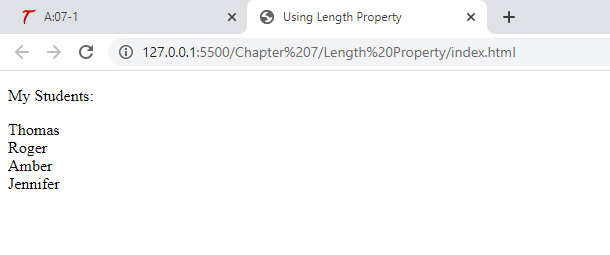
document.write(students[0] + "<br>");

document.write(students[1] + "<br>");

document.write(students[2] + "<br>");

document.write(students[3] + "<br>");

**Page 168 Using Length property**



<!DOCTYPE html>

<html>

<head>

    <title>Using Length Property</title>

</head>

<body>

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

</body>

</html>

let students = ["Thomas", "Roger", "Amber", "Jennifer"];

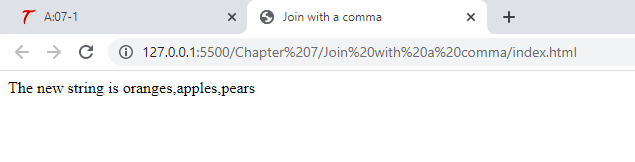
document.write("<p>My Students:</p>");

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

    document.write(students[i] + "<br>");

}

**Page 175 Join with a comma**

****

<!DOCTYPE html>

<html>

<head>

    <title>Join with a comma</title>

</head>

<body>

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

</body>

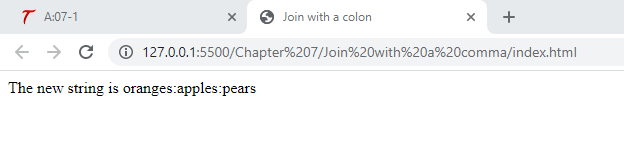
</html>

let fruits = ["oranges", "apples", "pears"];

let fruit\_string = fruits.join(",");

document.write("The new string is " + fruit\_string);

**Page 175 Join with a colon**

****

<!DOCTYPE html>

<html>

<head>

    <title>Join with a colon</title>

</head>

<body>

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

</body>

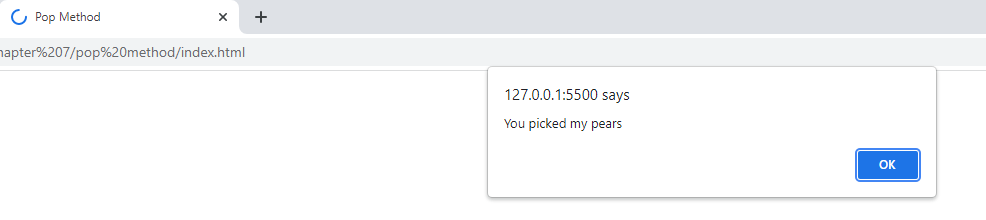
</html>

let fruits = ["oranges", "apples", "pears"];

let fruit\_string = fruits.join(":");

document.write("The new string is " + fruit\_string);

**Page 176 pop()**

****

<!DOCTYPE html>

<html>

<head>

    <title>Pop Method</title>

</head>

<body>

    <script src="popjs.js"></script>

</body>

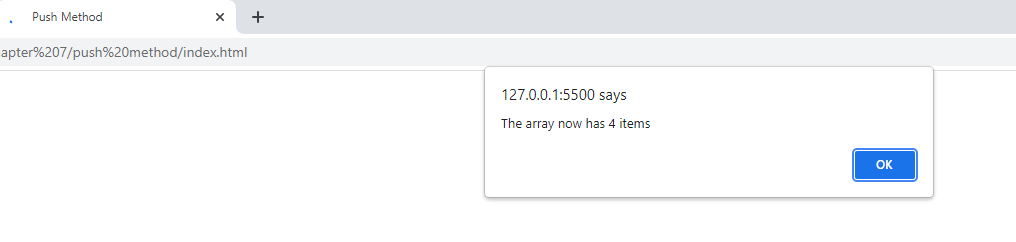
</html>

let fruits = ["oranges", "apples", "pears"];

let picket\_fruit = fruits.pop();

window.alert("You picked my " + picket\_fruit);

**Page 177 push()**

****

<!DOCTYPE html>

<html>

<head>

    <title>Push Method</title>

</head>

<body>

    <script src="pushjs.js"></script>

</body>

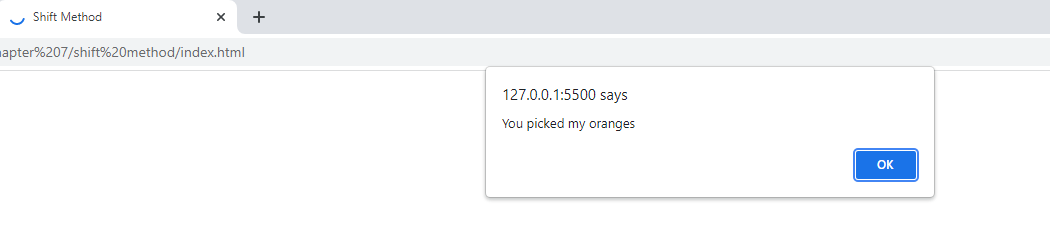
</html>

let fruits = ["oranges", "apples"];

let new\_length = fruits.push("pears", "grapes");

window.alert("The array now has " + new\_length + " items");

**Page 177 shift()**

****

<!DOCTYPE html>

<html>

<head>

    <title>Shift Method</title>

</head>

<body>

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

</body>

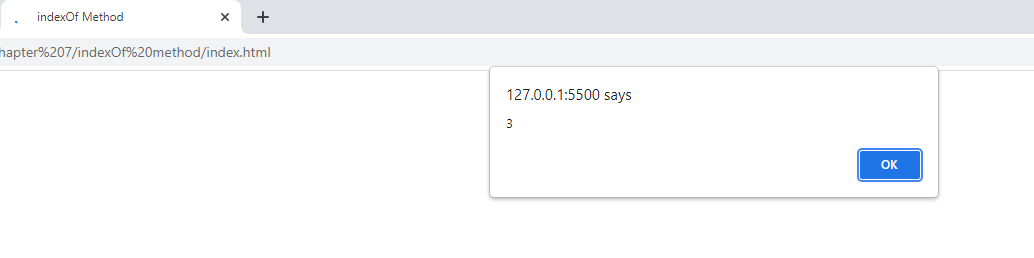
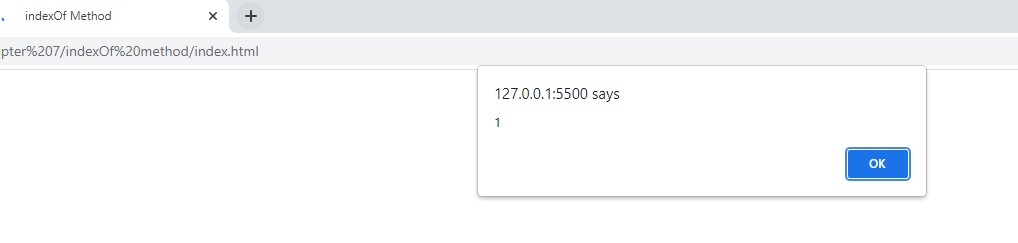
</html>

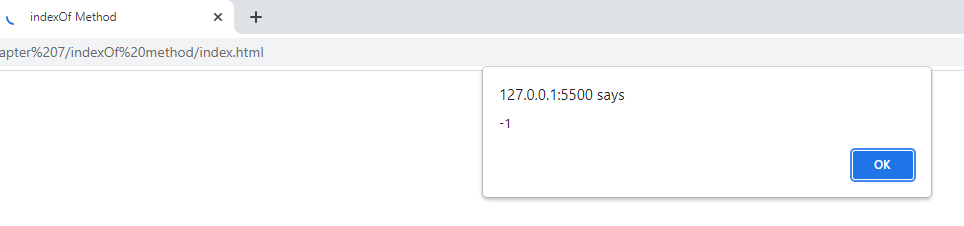
let fruits = ["oranges", "apples", "pears"];

let picked\_fruit = fruits.shift();

window.alert("You picked my " + picked\_fruit);

**Page 182 indexOf()**

****

****

<!DOCTYPE html>

<html>

<head>

    <title>indexOf Method</title>

</head>

<body>

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

</body>

</html>

let fruits = ["oranges", "apples", "pears", "apples"];

let found\_apples = fruits.indexOf("apples");

window.alert(found\_apples);

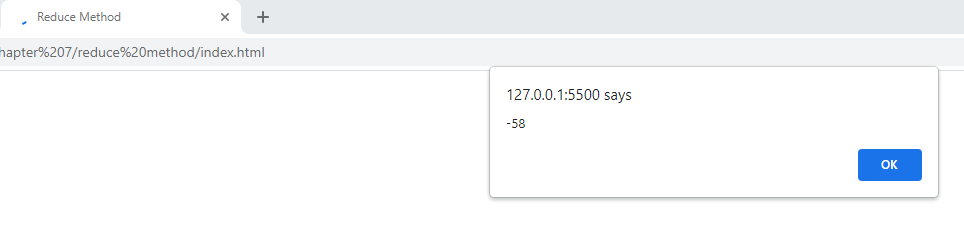
found\_apples = fruits.lastIndexOf("apples");

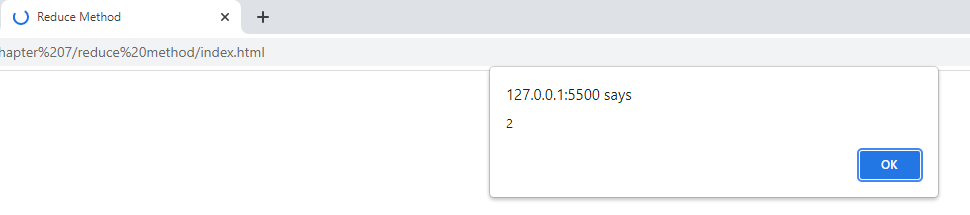
window.alert(found\_apples);

found\_apples = fruits.indexOf("grapes");

window.alert(found\_apples);

**Page 185 reduce()**

****

****

<!DOCTYPE html>

<html>

<head>

    <title>Reduce Method</title>

</head>

<body>

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

</body>

</html>

function sub\_values(prev\_value, next\_value) {

    return (prev\_value - next\_value);

}

let nums = [2, 4, 8, 16, 32];

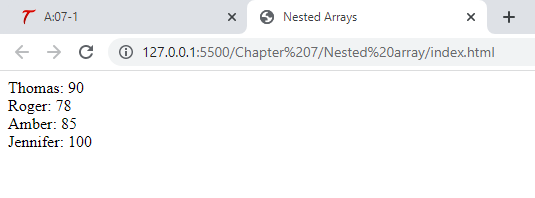
let the\_result = nums.reduce(sub\_values);

window.alert(the\_result);

the\_result = nums.reduceRight(sub\_values);

window.alert(the\_result);

**Page 190 Nested Array**

****

<!DOCTYPE html>

<html>

<head>

    <title>Nested Arrays</title>

</head>

<body>

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

</body>

</html>

let students = [

    ["Thomas", 92, 90],

    ["Roger", 87, 78],

    ["Amber", 81, 85],

    ["Jennifer", 99, 100]

];

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

    for (let j = 0; j < students[i].length; j++) {

        if (j === 0) {

            document.write(students[i][j] + ": ");

        } else if (j === students[i].length - 1) {

            document.write(students[i][j] + "<br>");

        } else {

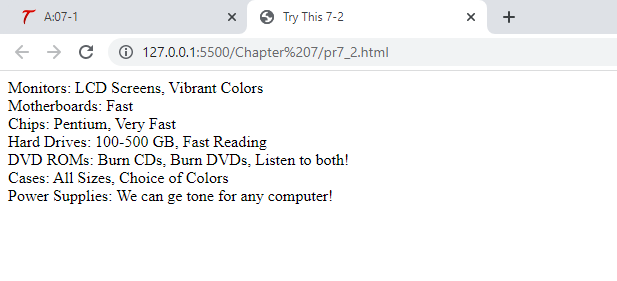
            continue;

        }

    }

}

**Page 192 Try This 7-2**

****

<!DOCTYPE html>

<html>

<head>

    <title>Try This 7-2</title>

</head>

<body>

    <script src="prjs7\_2.js"></script>

</body>

</html>

let computer\_parts = [

    ["Monitors", "LCD Screens", "Vibrant Colors"],

    ["Motherboards", "Fast"],

    ["Chips", "Pentium", "Very Fast"],

    ["Hard Drives", "100-500 GB", "Fast Reading"],

    ["DVD ROMs", "Burn CDs", "Burn DVDs", "Listen to both!"],

    ["Cases", "All Sizes", "Choice of Colors"],

    ["Power Supplies", "We can ge tone for any computer!"]

];

for (let i = 0; i < computer\_parts.length; i++) {

    for (let j = 0; j < computer\_parts[i].length; j++) {

        if (j === 0) {

            document.write(computer\_parts[i][j] + ": ");

        } else if (j === computer\_parts[i].length - 1) {

            document.write(computer\_parts[i][j] + "<br>");

        } else {

            document.write(computer\_parts[i][j] + ", ");

        }

    }

}