Practical – 1

Aim :

I. Write a JS arrow function to find missing number in an Array.

II. Write a JS arrow function to find second largest and second smallest number from given array.

i. **Code:**

```
<!DOCTYPE html>
<html lang="en">
<head>
    <meta charset="UTF-8">
    <meta name="viewport" content="width=device-width, initial-scale=1.0">
    <title>Document</title>
</head>
<body>
    <script>
        var arr = [1, 2, 4, 5, 6, 8, 10];
        const findMissingElement = (array) => {
            for (let i = 0; i < array.length; i++) {
                if (array[i] !== i + 1) {
                    return i + 1;
                }
            }
            return null;
        };
        console.log(arr);
        console.log(findMissingElement(arr));
    </script>
```

```
</body>
</html>
```

Output:

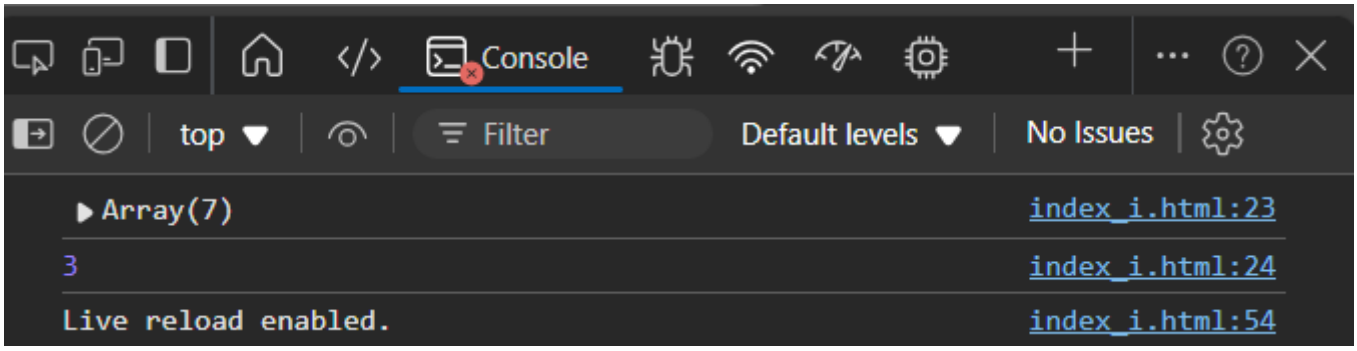

Figure I: *Missing number in Array*

ii. **Code:**

```
<!DOCTYPE html>
<html lang="en">
<head>
    <meta charset="UTF-8">
    <meta name="viewport" content="width=device-width, initial-scale=1.0">
    <title>Document</title>
</head>
<body>
    <script>
        var sort = (arr) => {
            for (var i = 0; i < arr.length; i++) {
                var min = i;
                for (var j = i + 1; j < arr.length; j++) {
                    if (arr[j] < arr[min]) {
                        min = j;
                    }
                }
                if (arr[min] < arr[i]) {
                    var temp = arr[min];
                    arr[min] = arr[i];
                    arr[i] = temp;
                }
            }
        };
        var array = [10, 8, 3, 6, 5, 1, 2];
        console.log(`Unsorted Array: ${array}`);
        sort(array);
        console.log(`Sorted Array: ${array}`);
        if (array.length >= 2) {
            console.log(`2nd Smallest Element: ${array[1]}`);
```

```
            console.log(`2nd Largest Element: ${array[array.length - 2]}`);
        } else {
            console.log("Array does not have enough elements to determine the 2nd smallest and 2nd largest.");
        }
    </script>
</body>
</html>
```

Output:

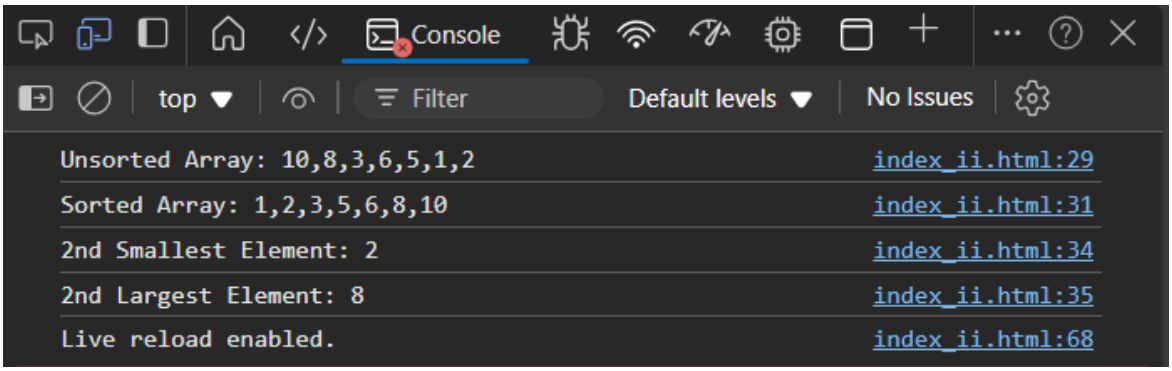

Figure II: *Second largest and second smallest number from given array.*

Practical – 2

Aim : Build a “Guess the Number Game” using JavaScript.

Code:

Index.html

```
<!DOCTYPE html>
<html lang="en">
<head>
    <meta charset="UTF-8">
    <meta name="viewport" content="width=device-width, initial-scale=1.0">
    <title>Number guessing</title>
    <style>
        * {
            margin: 0;
            padding: 0;
            box-sizing: border-box;
        }
        .container {
            border: 1px solid black;
            height: 400px;
            width: 400px;
            text-align: center;
            padding-top: 50px;
        }
        input {
            width: 200px;
            height: 50px;
            margin: 10px 0;
        }
```

```
        button {
            margin: 10px;
        }
        .msg {
            margin-top: 20px;
            font-size: 18px;
            color: red;
        }
    </style>
</head>
<body>
    <div class="container">
        <div class="input">
            <input type="number" id="guessInput" placeholder="Enter a number">
        </div>
        <div class="btn">
            <button onclick="submit()">Submit</button>
            <button onclick="start()">Start</button>
            <button onclick="again()">Play Again</button>
        </div>
        <div class="msg">
            <p id="msg"></p>
        </div>
    </div>

    <script>
        // Initialize tries and the secret number
        let tries = 5;
        let secretNumber;
        let gameStarted = false;
```

```
let msg = document.getElementById("msg");

// Function to generate random number between 1 and 100
const gen = () => {
    return Math.floor(Math.random() * 100) + 1;
};

// Function to reset the game
const again = () => {
    tries = 5;
    gameStarted = false;
    msg.innerHTML = "Game reset. Please click 'Start' to begin.";
    document.getElementById("guessInput").value = "";  // Clear the input field
    document.getElementById("guessInput").disabled = false;  // Re-enable the input field
    document.querySelector("button:nth-child(2)").innerHTML = "Start";  // Change button text to "Start"
};

// Function to start the game
const start = () => {
    let button = document.querySelector("button:nth-child(2)"); // "Start" button
    if (!gameStarted) {
        secretNumber = gen();
        gameStarted = true;
        tries = 5;
        button.innerHTML = "Stop";
        msg.innerHTML = `Game Started! You have ${tries} tries.`;
    } else {
        gameStarted = false;
        button.innerHTML = "Start";
        msg.innerHTML = "Game over!";
```

```
        document.getElementById("guessInput").disabled = true;
    }
};

// Function to handle submit guess
const submit = () => {
    if (!gameStarted) {
        msg.innerHTML = "Please start the game first!";
        return;
    }

    let guess = parseInt(document.getElementById("guessInput").value);
    if (isNaN(guess)) {
        msg.innerHTML = "Please enter a valid number.";
        return;
    }

    // Check if guess is correct
    if (guess === secretNumber) {
        msg.innerHTML = `Congratulations! You guessed the number ${secretNumber} correctly!`;
        document.getElementById("guessInput").disabled = true;
    } else {
        tries--;
        if (guess < secretNumber) {
            msg.innerHTML = `Incorrect! Guess higher. You have ${tries} tries left.`;
        } else if (guess > secretNumber) {
            msg.innerHTML = `Incorrect! Guess lower. You have ${tries} tries left.`;
        }

        if (tries <= 0) {
```

```
            msg.innerHTML = `Game over! The correct number was ${secretNumber}.`;
            document.getElementById("guessInput").disabled = true;
        }
    }

    document.getElementById("guessInput").value = "";
};

</script>
</body>
</html>
```

Fig 2.1 : normal setup

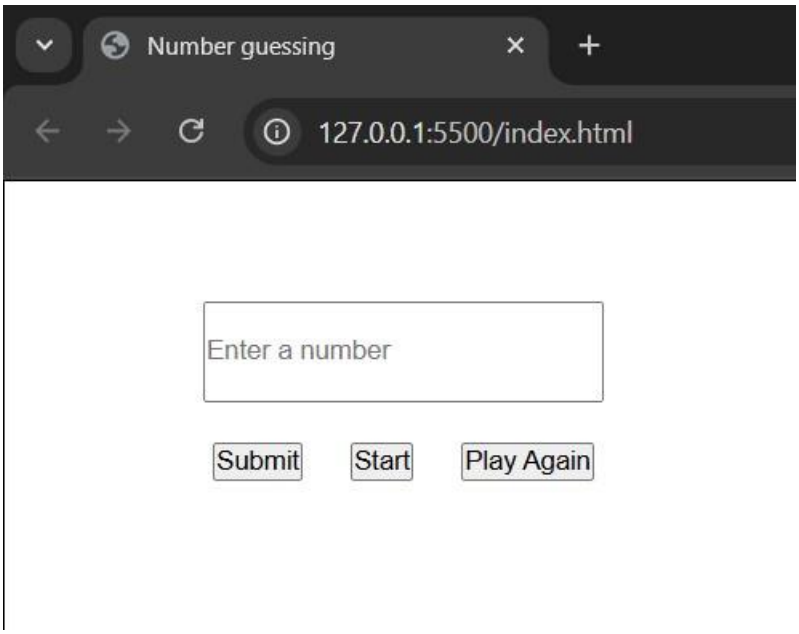

fig 2.2 : starting the game

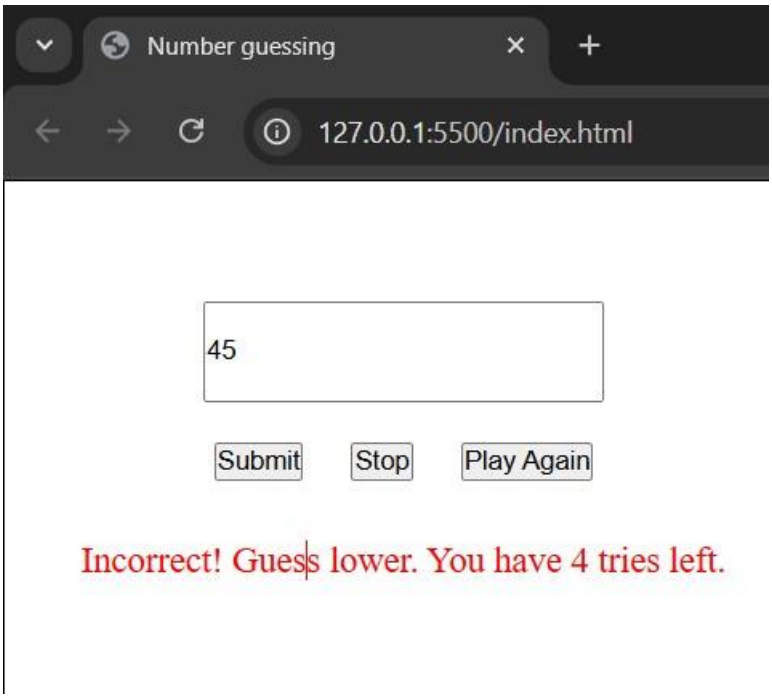

Fig 2.3 : after finishing the game