**Practical Task**

Q1. Completed

Q2. Completed

**Note:** Validation for name filed is not added as it is not mentioned in the requirement.

**Theoretical Questionnaire — [JavaScript]**

Q1. What is the difference between Cookie / Local Storage / Session Storage?

**Answer:**

**Cookie:**

1. It stores data that has to be sent back to the server with subsequent requests. Its expiration varies based on the type of cookie whereas the expiration duration can be set from either server-side or client-side (normally from server-side).

2. Storage limit is the lowest amongst Cookie, LocalStorage and SessionStorage.

3. Cookies are primarily for server-side reading (can also be read on client-side).

**LocalStorage:**

1. It stores data with no expiration date and gets cleared only through JavaScript or clearing the browser cache/ locally stored data.

2. Storage limit is the highest amongst Cookie, LocalStorage and SessionStorage.

3. LocalStorage can only be read on client-side.

**SessionStorage:**

1. The sessionStorage object stores data only for a session, which means that the data is stored until the browser window or tab is closed. Data is never transferred to the server.

2. The storage limit is higher than that of cookie but lower than LocalStorage.

3. SessionStorage can only be read on client-side.

Q2. What is the significance of, and reason for, wrapping the entire content of a JavaScript source

file in a function block?

**Answer:** The entire content of a JavaScript source file is wrapped in a function block mainly to avoid polluting the global scope and overriding already existing variables.

Q3. What will the code below output? Explain your answer.

1. console.log(0.1 + 0.2);
2. console.log(0.1 + 0.2 == 0.3);

**Answer:**

1. 0.30000000000000004
2. False

Because JavaScript numbers are always stored as double precision floating point numbers (64-bit), following the international IEEE 754 standard.

Q4. Write a sum method which will work properly when invoked using either syntax below.

console.log(sum(2,3)); // Outputs 5

console.log(sum(2)(3)); // Outputs 5

**Answer:**

function sum(a,b){

if(b===undefined){

return function(c){ return parseInt(a)+parseInt(c);}

}

return parseInt(a)+parseInt(b);

}

Q5. What is the output of the following code?

var length = 10;

function fn() {

console.log(this.length);

}

var obj = {

length: 5,

method: function(fn) {

fn();

arguments[0]();

}

};

obj.method(fn, 1);

**Answer:** 10, 2