

**Lab 06**

### CSC-325: Web Engineering

### Semester VI (CS, SE) Section (B) (Spring 2022)

### Course Instructor(s): Riaz Ali Soomro

### Sajjad Ali

### 023-19-0100

**1. Write a JavaScript function to get the values of First and Last name of the**

**following form (and show them in an alert dialog).**

<!DOCTYPE html>

<html>

<head>

<meta charset="utf-8" />

<title>Return first and last name from a form</title>

</head>

<body>

<form id="form1" onsubmit="getFormvalue()">

First name:

<input type="text" name="fname" value="David" /><br />

Last name:

<input type="text" name="lname" value="Beckham" /><br />

<input type="submit" value="Submit" />

</form>

<script>

**function** getFormvalue() {

**let** fname = document.getElementsByName("fname")[0].value;

**let** lname = document.getElementsByName("lname")[0].value;

console.log(fname + " " + lname);

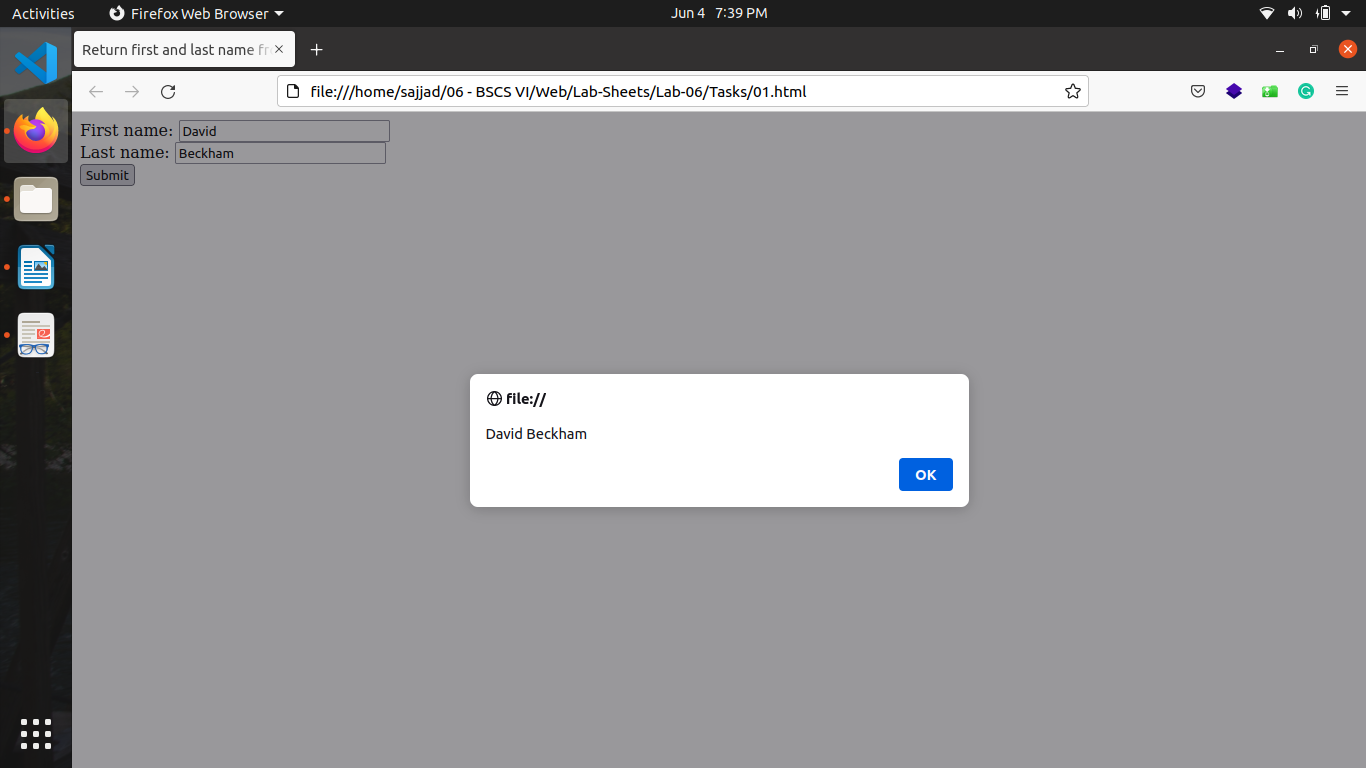
alert(fname + " " + lname);

}

</script>

</body>

</html>



**2. Write a JavaScript program to set the background color of a paragraph.**

<!DOCTYPE html>

<html>

<head>

<meta charset="utf-8" />

<title>JS DOM</title>

</head>

<body>

<input

type="button"

value="Click to set paragraph background color"

onclick="set\_background()"

/>

<p>w3resource JavaScript Exercises</p>

<p>w3resource PHP Exercises</p>

</body>

<script>

**function** set\_background() {

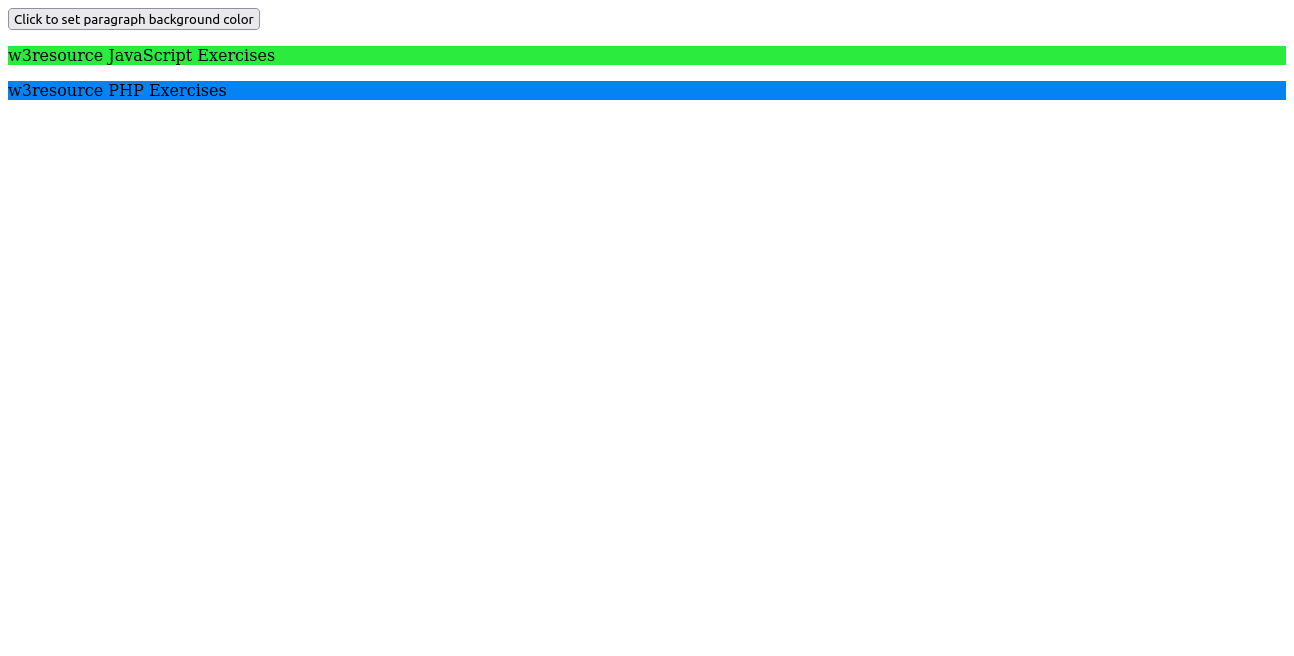
for (**let** p of document.getElementsByTagName("p"))

p.style.backgroundColor = `rgb(${Math.random(255) \* 255}, $ {Math.random(255) \* 255}, ${Math.random(255) \* 255})`;

}

</script>

</html>



**3. Here is a sample html file with a submit button. Write a JavaScript function to get the value of the href, hreflang, rel, target, and type attributes of the specified link.**

<!DOCTYPE html>

<html>

<head>

<meta charset="utf-8" />

</head>

<body>

<p>

<a

id="w3r"

type="text/html"

hreflang="en-us"

rel="nofollow"

target="\_self"

href="http://www.w3resource.com/"

>w3resource</a>

</p>

<button onclick="getAttributes()">Click here to get attributes value

</button>

<script>

function getAttributes() {

let anchor = document.getElementById("w3r");

alert(

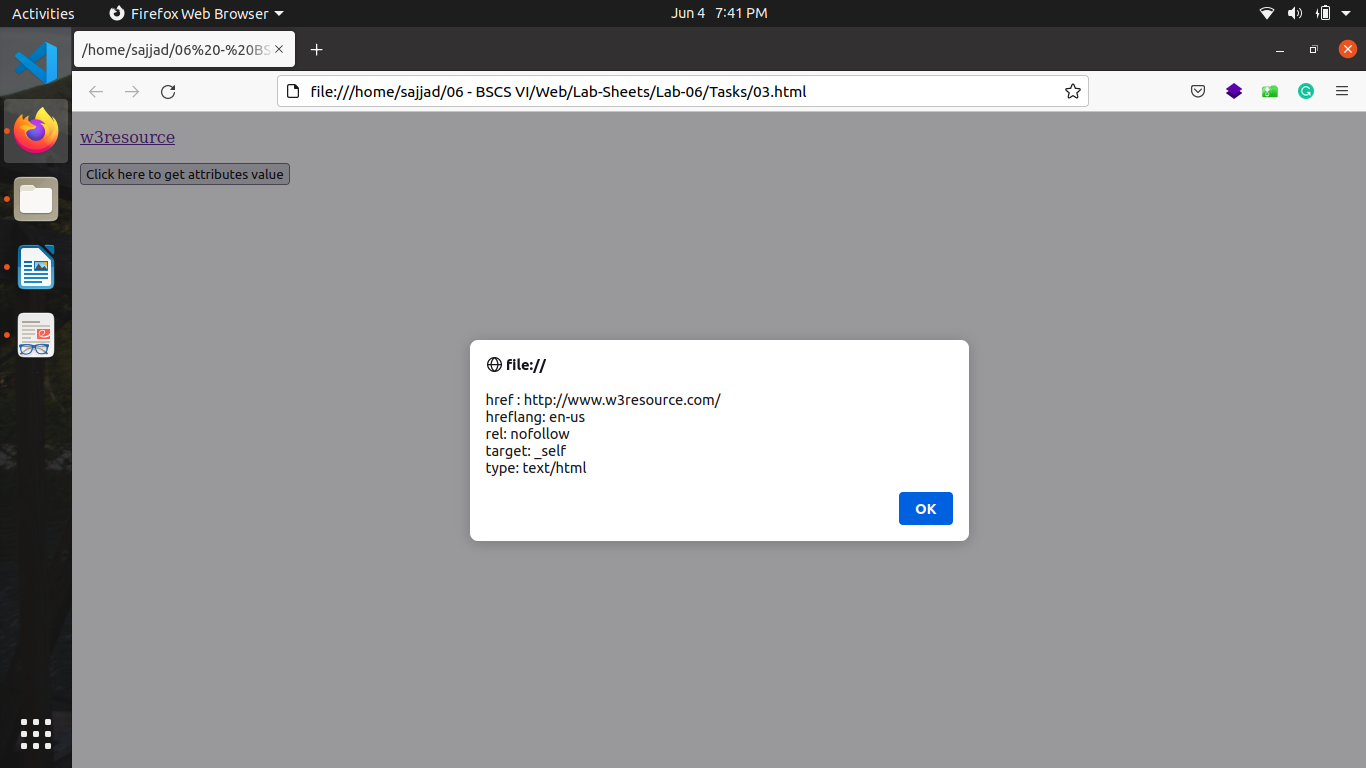
`href : ${anchor.href} \nhreflang: ${anchor.hreflang} \ nrel: ${anchor.rel} \ntarget: ${anchor.target} \ntype: $ {anchor.type}`);

}

</script>

</body>

</html>



**4. Here is a sample html file with a submit button. Now modify the style of the**

**paragraph text (such as fontSize, fontFamily, color, etc. ) through javascript**

**code.**

<!DOCTYPE html>

<html>

<head>

<meta charset="utf-8" />

<title>JS DOM paragraph style</title>

</head>

<body>

<p id="text">JavaScript Exercises - w3resource</p>

<div>

<button id="jsstyle" onclick="js\_style()">Style</button>

</div>

<script>

function js\_style() {

let p = document.getElementById("text").style;

p.color = "blue";

p.backgroundColor = "cyan";

}

</script>

</body>

</html>



**5. Write a JavaScript function to add rows to a table.**

<!DOCTYPE html>

<html>

<head>

<meta charset="utf-8" />

<title>Insert row in a table - w3resource</title>

</head>

<body>

<table id="sampleTable" border="1">

<tr>

<td>Row1 cell1</td>

<td>Row1 cell2</td>

</tr>

<tr>

<td>Row2 cell1</td>

<td>Row2 cell2</td>

</tr>

</table>

<br />

<input type="button" onclick="insert\_Row()" value="Insert row" />

<script>

let row = 3;

function insert\_Row() {

let table = document.getElementById("sampleTable");

let tr = document.createElement("tr");

for (let i = 1; i <= 2; i++) {

let td = document.createElement("td");

td.innerText = `Row${row} cell${i}`;

tr.appendChild(td);

}

table.appendChild(tr);

row++;

}

</script>

</body>

</html>



**6. Given the following HTML. Write the code necessary to do the following:**

**1. Select the section with an id of container without using querySelector .**

**2. Select the section with an id of container using querySelector .**

**3. Select all of the list items with a class of "second".**

**4. Select a list item with a class of third, but only the list item inside of**

**the ol tag.**

**5. Give the section with an id of container the text "Hello!".**

**6. Add the class main to the div with a class of footer .**

**7. Remove the class main on the div with a class of footer .**

**8. Create a new li element.**

**9. Give the li the text "four".**

**10. Append the li to the ul element.**

**11. Loop over all of the li s inside the ol tag and give them a background color**

**of "green".**

**12. Remove the div with a class of footer .**

// html code

<!DOCTYPE html>

<html lang="en">

<head>

<meta charset="UTF-8" />

<title>Document</title>

</head>

<body>

<div class="header"></div>

<section id="container">

<ul>

<li class="first">one</li>

<li class="second">two</li>

<li class="third">three</li>

</ul>

<ol>

<li class="first">one</li>

<li class="second">two</li>

<li class="third">three</li>

</ol>

</section>

<div class="footer">Footer</div>

<script src="06.js"></script>

</body>

</html>

// javascript file 06.js

onload = function(){

// 1

let section1 = this.document.getElementById('container');

//2

let section2 = this.document.querySelector('section#container');

// 3

let listItemInClassSecond = this.document.getElementsByClassName('second');

//4

let listItemInClassThird = this.document.querySelector('ol .third');

// 5

section1.append('Hello!');

// 6

this.document.querySelector('div.footer').classList.add('main');

// 7

this.document.querySelector('div.footer').classList.remove('main');

// 8

let li = this.document.createElement('li');

// 9

li.innerText = 'Four';

// 10

this.document.querySelector('ul').appendChild(li);

// 11

let listItemsInOl = this.document.querySelectorAll('ol li');

for(let li of listItemsInOl)

li.style.backgroundColor = 'green';

// 12

this.document.body.removeChild(this.document.querySelector('div.footer'));

}



**7. Given the following HTML, create a script.js file to complete the first two**

**parts.**

**1. Add the necessary code to wait for the DOM to load to make sure that**

**anything you manipulate in the DOM has loaded. You can do this either**

**using window.onload or adding an event listener for**

**DOMContentLoaded.**

**2. Replace the text "Change me" with "Hello World!".**

**3. When a user hovers over one of the colored boxes change the text to**

**display the color that is being hovered over.**

**4. Create a new div element.**

**5. Give your new div a class of purple and style it so that it has a**

**background color of purple.**

**6. Append your new div to the page to the section tag.**

**// html code**

<!DOCTYPE html>

<html lang="en">

<head>

<meta charset="UTF-8" />

<title>DOM Exercise</title>

<style>

div {

width: 50px;

height: 50px;

display: inline-block;

}

.brown {

background-color: brown;

}

.green {

background-color: green;

}

.blue {

background-color: blue;

}

.purple {

background-color: purple;

}

.yellow {

background-color: yellow;

}

.car1 {

background-color: #8c9c12;

}

.car2 {

background-color: #1da788;

}

.car1,

.car2 {

margin-left: 0;

}

</style>

</head>

<body>

<h1 id="change\_heading">Change Me!</h1>

SELECTED COLOR <span class="selected">None!</span>

<section>

<div class="brown"></div>

<div class="green"></div>

<div class="blue"></div>

<div class="yellow"></div>

</section>

<h2>Race!</h2>

<button>Start the race!</button>

<br />

<div class="car1"></div>

<br />

<div class="car2"></div>

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

</body>

</html>

**// javascript code**

onload = function(){

// 2

document.querySelector('#change\_heading').innerText = 'Hello World!';

// 3

document.querySelector('div.brown').addEventListener('mouseover', changeTextAsColor);

document.querySelector('div.brown').addEventListener('mouseleave', writeNone);

document.querySelector('div.green').addEventListener('mouseover', changeTextAsColor);

document.querySelector('div.green').addEventListener('mouseleave', writeNone);

document.querySelector('div.blue').addEventListener('mouseover', changeTextAsColor);

document.querySelector('div.blue').addEventListener('mouseleave', writeNone);

document.querySelector('div.yellow').addEventListener('mouseover', changeTextAsColor);

document.querySelector('div.yellow').addEventListener('mouseleave', writeNone);

// 4

let div = this.document.createElement('div');

// 5

div.classList.add('purple');

div.addEventListener('mouseover', changeTextAsColor);

div.addEventListener('mouseleave', writeNone);

// 6

this.document.querySelector('section').appendChild(div);

}

// 3

function changeTextAsColor(event){

document.querySelector('span.selected').innerText = event.target.className;

}

function writeNone(){

document.querySelector('span.selected').innerText = 'None';

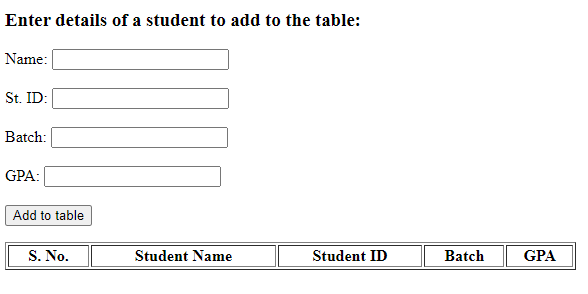
}

**8. Create an HTML page that should contain a few text fields to get input from**

**the user, a button captioned “Add to table”, and a table. Initially the table**

**should only contain a header row (no other data should be there) as shown**

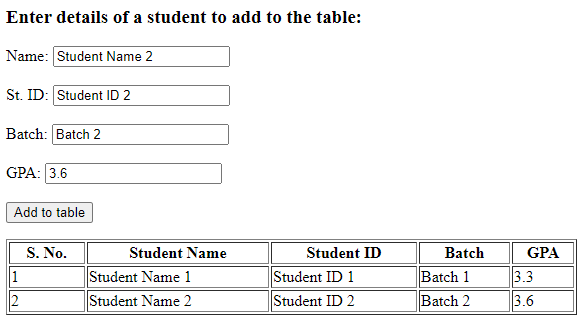
**below:**



**Then, whenever the user enters some data and presses the button, a new**

**row/record comprising the entered data should be added to the table**

**dynamically (using JavaScript) as shown in the sample below:**



<!DOCTYPE html>

<html lang="en">

<head>

<meta charset="UTF-8" />

<meta http-equiv="X-UA-Compatible" content="IE=edge" />

<meta name="viewport" content="width=device-width, initial-scale=1.0" />

<title>Document</title>

</head>

<body>

<h2>Enter details of a student to add in table</h2>

<table>

<tr>

<td><label for="name">Name : </label></td>

<td><input type="text" id="name" /></td>

</tr>

<tr>

<td><label for="id">St. ID : </label></td>

<td><input type="text" id="id" /></td>

</tr>

<tr>

<td><label for="gpa">GPA : </label></td>

<td><input type="text" id="gpa" /></td>

</tr>

<tr>

<td><label for="batch">Batch : </label></td>

<td><input type="text" id="batch" /></td>

</tr>

<br />

<tr>

<td colspan="2">

<button type="button" onclick="add()">Add to table</button>

</td>

</tr>

</table>

<br />

<table border="1" id="records">

<tr>

<th>S. No</th>

<th>Student Name</th>

<th>Student ID</th>

<th>Batch</th>

<th>GPA</th>

</tr>

</table>

<script>

let record = 0;

function add() {

let data = [

++record,

document.querySelector("#name").value,

document.querySelector("#id").value,

document.querySelector("#batch").value,

document.querySelector("#gpa").value,

];

let table = document.querySelector("#records");

let tr = document.createElement("tr");

for (let d of data) {

let td = document.createElement("td");

td.innerText = d;

tr.appendChild(td);

}

table.appendChild(tr);

}

</script>

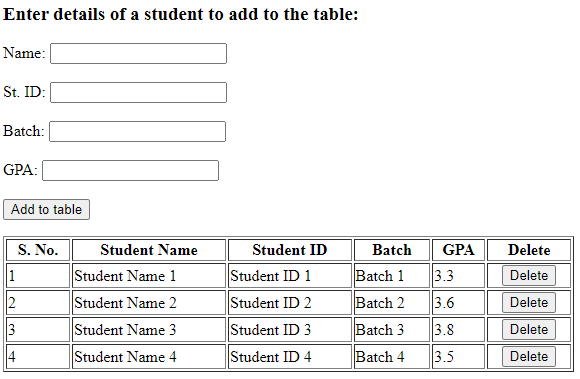
</body>

</html>

**9. Extend the functionality of the previous task. Now, there should be an**

**additional column in the table, where in every row, there should a button**

**captioned Delete, as shown below:**



**When the user clicks the Delete button, then that entire row should be deleted**

**(in which row that button was present). For example, when the user clicks the**

**delete button in the second row, then second row should be deleted and table**

**should be updated, as shown below:**

<!DOCTYPE html>

<html lang="en">

<head>

<meta charset="UTF-8" />

<meta http-equiv="X-UA-Compatible" content="IE=edge" />

<meta name="viewport" content="width=device-width, initial-scale=1.0" />

<title>Document</title>

</head>

<body>

<h2>Enter details of a student to add in table</h2>

<table>

<tr>

<td><label for="name">Name : </label></td>

<td><input type="text" id="name" /></td>

</tr>

<tr>

<td><label for="id">St. ID : </label></td>

<td><input type="text" id="id" /></td>

</tr>

<tr>

<td><label for="gpa">GPA : </label></td>

<td><input type="text" id="gpa" /></td>

</tr>

<tr>

<td><label for="batch">Batch : </label></td>

<td><input type="text" id="batch" /></td>

</tr>

<br />

<tr>

<td colspan="2">

<button type="button" onclick="add()">Add to table</button>

</td>

</tr>

</table>

<br />

<table border="1" id="records">

<tr>

<th>S. No</th>

<th>Student Name</th>

<th>Student ID</th>

<th>Batch</th>

<th>GPA</th>

<th>Delete</th>

</tr>

</table>

<script>

let record = 0;

function add() {

let data = [

++record,

document.querySelector("#name").value,

document.querySelector("#id").value,

document.querySelector("#batch").value,

document.querySelector("#gpa").value,

];

let table = document.querySelector("#records");

let tr = document.createElement("tr");

for (let d of data) {

let td = document.createElement("td");

td.innerText = d;

tr.appendChild(td);

}

let button = document.createElement("button");

button.setAttribute("name", record);

button.innerText = "delete";

button.onclick = deleteRecord;

tr.appendChild(button);

table.appendChild(tr);

}

function deleteRecord(event) {

let serialNo = event.target.name;

let table = document.querySelector("#records");

let tableRows = document.querySelectorAll("#records tr");

for (let tr of tableRows) {

if (tr.children[0].innerText == serialNo) {

table.removeChild(tr);

break;

}

}

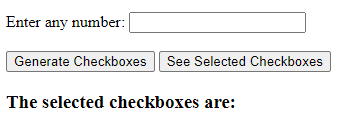
}

</script>

</body>

</html>

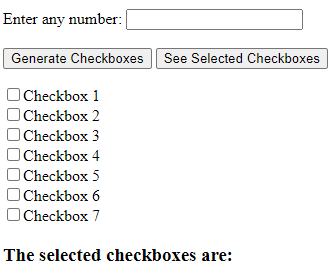
**10. Create an HTML page that should initially contain a text field and two buttons as shown below:**

**The user should enter a number indicating how many checkboxes he/she wants**

**to create/generate. And then when the “Generate Checkboxes” button is**

**clicked, that much number of checkboxes should be created dynamically at**

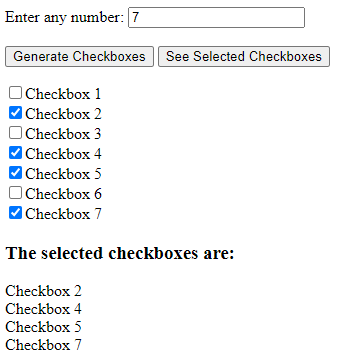
**runtime (using JavaScript) as shown in the sample given below.**

****

**Besides, there should also be another button with the caption “See Selected**

**Checkboxes”, and when it is clicked, you should tell which checkboxes are**

**checked/selected (using JavaScript).**

****

<!DOCTYPE html>

<html lang="en">

<head>

<meta charset="UTF-8" />

<meta http-equiv="X-UA-Compatible" content="IE=edge" />

<meta name="viewport" content="width=device-width, initial-scale=1.0" />

<title>Document</title>

</head>

<body>

<table>

<tr>

<td><label for="number">Enter any number : </label></td>

<td><input type="text" id="number" /></td> </tr>

<tr>

<td>

<button type="button" onclick="generate()">

Generate Checkboxes

</button>

</td>

<td>

<button type="button" onclick="view()">

See Selected Checkboxes

</button>

</td>

</tr>

</table>

<div id="checkboxes"></div>

<h2>The selected checkboxes are :</h2>

<div id="selected"></div>

<script>

function generate() {

document.querySelector("#selected").innerText = "";

let div = document.getElementById("checkboxes");

div.innerHTML = "";

let num = Number(document.getElementById("number").value);

for (let i = 1; i <= num; i++) {

let label = document.createElement("label");

label.setAttribute("for", `${i}`);

label.innerText = `Checkbox${i}`;

let cb = document.createElement("input");

cb.setAttribute("id", `${i}`);

cb.setAttribute("type", "checkbox");

div.appendChild(cb);

div.appendChild(label);

div.appendChild(document.createElement("br"));

}

}

function view() {

let checkboxes = document.querySelectorAll("#checkboxes input");

let div = document.querySelector("#selected");

div.innerText = "";

let i = 1;

for (let cb of checkboxes) {

if (cb.checked) div.innerText += `CheckBox${i}`;

i++;

}

}

</script>

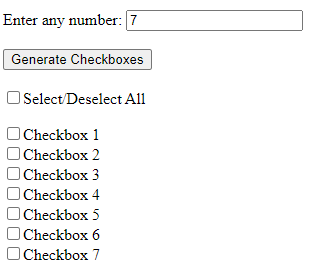
</body>

</html>

**11. Create a task similar to the previous one, where user should enter a number**

**and that much number of checkboxes should be created when the button is**

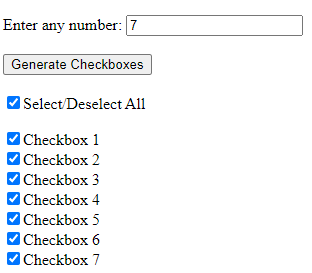
**clicked, as shown in the sample below:**

****

**In addition, there should be a checkbox with the label/caption “Select/Deselect**

**All”, and when that checkbox is selected/checked, then all the other**

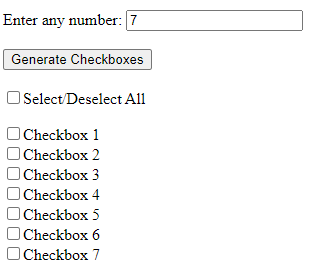
**checkboxes should also be automatically selected/checked, as shown below:**

****

**In the same way, when that checkbox is deselected/unchecked, then all the**

**other checkboxes should be automatically deselected/unchecked, as shown**

**below:**



<!DOCTYPE html>

<html lang="en">

<head>

<meta charset="UTF-8" />

<meta http-equiv="X-UA-Compatible" content="IE=edge" />

<meta name="viewport" content="width=device-width, initial-scale=1.0" />

<title>Document</title>

</head>

<body>

<table>

<tr>

<td><label for="number">Enter any number : </label></td>

<td><input type="text" id="number" /></td>

</tr>

<tr>

<td>

<button type="button" onclick="generate()">

Generate Checkboxes

</button>

</td>

</tr>

</table>

<br /><br />

<input type="checkbox" id="selectDeselect" onclick="selectAll()" />

<label for="selectDeselect">Select/Deselect All</label> <br /><br />

<br /><br />

<div id="checkboxes"></div>

<script>

function generate() {

let div = document.getElementById("checkboxes");

div.innerHTML = "";

let num = Number(document.getElementById("number").value);

for (let i = 1; i <= num; i++) {

let cb = document.createElement("input");

cb.setAttribute("id", `${i}`);

cb.setAttribute("type", "checkbox");

let label = document.createElement("label");

label.setAttribute("for", `${i}`);

label.innerText = `Checkbox${i}`;

div.appendChild(cb);

div.appendChild(label);

div.appendChild(document.createElement("br"));

}

}

function selectAll(event) {

let checkboxes = document.querySelectorAll("#checkboxes input");

let value = false;

if (document.getElementById("selectDeselect").checked) value = true;

for (let c of checkboxes) c.checked = value;

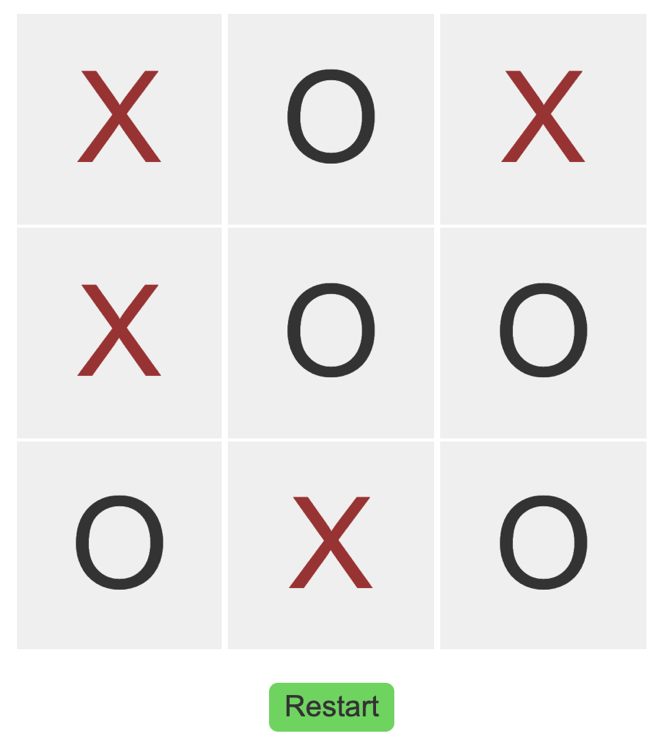
}

</script>

</body>

</html>

**12. Create a Tic-Tac-Toe game with two players. Following is a sample output.**

****

<!DOCTYPE html>

<html lang="en">

<head>

<meta charset="UTF-8" />

<meta http-equiv="X-UA-Compatible" content="IE=edge" />

<meta name="viewport" content="width=device-width, initial-scale=1.0" />

<title>Document</title>

<style>

td {

display: inline-flex;

justify-content: center;

align-items: center;

width: 200px;

height: 100px;

background-color: lightgray;

font-weight: bolder;

color: brown;

}

</style>

</head>

<body>

<table>

<tr>

<td></td>

<td></td>

<td></td>

</tr>

<tr>

<td></td>

<td></td>

<td></td>

</tr>

<tr>

<td></td>

<td></td>

<td></td>

</tr>

</table>

<button type="button" onclick="resetData()">Reset</button>

<script>

let x = true;

onload = function () {

let tds = document.querySelectorAll("td");

for (let t of tds) {

t.innerText = "";

t.onclick = playGame;

}

};

function playGame(event) {

// debugger;

if (x) {

event.target.innerText = "X";

event.target.style.color = "brown";

} else {

event.target.innerText = "O";

event.target.style.color = "black";

}

x = !x;

event.target.onclick = null;

let tr = document.querySelectorAll("tr");

let data = [[tr[0].children], [tr[1].children], [tr[2].children]];

if (

checkValues(

data[0][0][0].innerText,

data[0][0][1].innerText,

data[0][0][2].innerText,

"X"

) ||

checkValues(

data[1][0][0].innerText,

data[1][0][1].innerText,

data[1][0][2].innerText,

"X"

) ||

checkValues(

data[2][0][0].innerText,

data[2][0][1].innerText,

data[2][0][2].innerText,

"X"

) ||

checkValues(

data[0][0][0].innerText,

data[1][0][0].innerText,

data[2][0][0].innerText,

"X"

) ||

checkValues(

data[0][0][1].innerText,

data[1][0][1].innerText,

data[2][0][1].innerText,

"X"

) ||

checkValues(

data[0][0][2].innerText,

data[1][0][2].innerText,

data[2][0][2].innerText,

"X"

) ||

checkValues(

data[0][0][0].innerText,

data[1][0][1].innerText,

data[2][0][2].innerText,

"X"

) ||

checkValues(

data[2][0][0].innerText,

data[1][0][1].innerText,

data[0][0][2].innerText,

"X"

)

) {

alert("X Won");

document.querySelector("button").click();

} else if (

checkValues(

data[0][0][0].innerText,

data[0][0][1].innerText,

data[0][0][2].innerText,

"O"

) ||

checkValues(

data[1][0][0].innerText,

data[1][0][1].innerText,

data[1][0][2].innerText,

"O"

) ||

checkValues(

data[2][0][0].innerText,

data[2][0][1].innerText,

data[2][0][2].innerText,

"O"

) ||

checkValues(

data[0][0][0].innerText,

data[1][0][0].innerText,

data[2][0][0].innerText,

"O"

) ||

checkValues(

data[0][0][1].innerText,

data[1][0][1].innerText,

data[2][0][1].innerText,

"O"

) ||

checkValues(

data[0][0][2].innerText,

data[1][0][2].innerText,

data[2][0][2].innerText,

"O"

) ||

checkValues(

data[0][0][0].innerText,

data[1][0][1].innerText,

data[2][0][2].innerText,

"O"

) ||

checkValues(

data[2][0][0].innerText,

data[1][0][1].innerText,

data[0][0][2].innerText,

"O"

)

) {

alert("O Won");

document.querySelector("button").click();

}

}

function checkValues(a, b, c, s) {

return a == s && b == s && c == s;

}

function resetData() {

let tds = document.querySelectorAll("td");

for (let t of tds) {

t.innerText = "";

t.onclick = playGame;

}

x = true;

}

</script>

</body>

</html>

**13. For this task you will be combining your knowledge of DOM, events and the localStorage to build a To-Do app! In this app, a user, you should be able to:**

* ** Add a new to-do (by submitting a form)**
* ** Mark a to-do as completed (by striking through the text of the to-do)**
* ** Remove a to-do**

**Using the localStorage, try to store your to-dos so that if you refresh the page you do not lose what you have added to the list! Also, try to save to-dos that you have marked as completed!**

<!DOCTYPE html>

<html lang="en">

<head>

<meta charset="UTF-8" />

<meta http-equiv="X-UA-Compatible" content="IE=edge" />

<meta name="viewport" content="width=device-width, initial-scale=1.0" />

<title>Document</title>

</head>

<body>

<table>

<tr>

<td><label for="input">Enter a to do task</label></td>

</tr>

<tr>

<td><input type="text" id="input" /></td>

<td><button onclick="saveRecord()">Save</button></td>

</tr>

</table>

<br /><br />

<table id="list"></table>

<script>

let i = 0;

document.addEventListener("DOMContentLoaded", () => {

let keys = Object.keys(localStorage).sort(); for (let k of keys) {

addItem(k, localStorage.getItem(k));

i = k;

}

if (i) i++;

});

function saveRecord(event) {

let text = document.getElementById("input").value;

addItem(i, "0" + text);

localStorage.setItem(i, "0" + text);

i++;

}

function strike(event) {

let label = event.target.nextElementSibling;

if (!label.children.length) {

let s = document.createElement("s");

s.innerText = label.innerText;

label.innerHTML = "";

label.appendChild(s);

localStorage.setItem(label.getAttribute("for"), "1" + s.innerText);

} else {

let text = label.children[0].innerText;

label.innerHTML = "";

label.innerText = text;

localStorage.setItem(label.getAttribute("for"), "0" + s.innerText);

}

}

function deleteTask(event) {

localStorage.removeItem(event.target.getAttribute("name"));

event.target.parentElement.parentElement.remove();

}

function addItem(i, text) {

let section = document.getElementById("list");

let tr = document.createElement("tr");

let td = document.createElement("td");

let cb = document.createElement("input");

cb.setAttribute("type", "checkbox");

cb.setAttribute("id", i);

cb.onclick = strike;

td.appendChild(cb);

let label = document.createElement("label");

label.setAttribute("for", i);

if (text[0] == "0") label.innerText = text.substr(1);

else {

let s = document.createElement("s");

s.innerText = text.substr(1);

label.appendChild(s);

cb.checked = true;

}

td.appendChild(label);

tr.appendChild(td);

td = document.createElement("td");

let btn = document.createElement("a");

btn.setAttribute("name", i);

btn.setAttribute("href", "//");

btn.onclick = deleteTask;

btn.innerText = "Delete";

td.appendChild(btn);

tr.appendChild(td);

section.append(tr);

}

</script>

</body>

</html>

**14. Using the localStorage, create a page that shows the count/number of times that page has been visited till now. Every time the page is refreshed or visited, the count should be incremented by one to indicate the updated number of**

**visits.**

<!DOCTYPE html>

<html lang="en">

<head>

<meta charset="UTF-8" />

<meta http-equiv="X-UA-Compatible" content="IE=edge" />

<meta name="viewport" content="width=device-width, initial-scale=1.0" />

<title>Document</title>

<style>

div {

width: 200px;

height: 50px;

background-color: burlywood;

display: flex;

justify-content: center;

align-items: center;

}

</style>

</head>

<body>

<div id="clicks">0 Clicks</div>

<script>

let i = 0;

document.addEventListener("DOMContentLoaded", () => {

if (localStorage.getItem("clicks")) i = +localStorage.getItem("clicks");

localStorage.setItem("clicks", ++i);

document.getElementById("clicks").innerText = i + " Clicks";

});

</script>

</body>

</html>