**Solutions-Lab: DOM Manipulations**

1. **List of Items**

function addItem(){

let inputElement = document.getElementById('newItemText');

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

li.textContent = inputElement.value;

let ul = document.getElementById('items');

ul.appendChild(li);

inputElement.value = '';

}

<h1>List of Items</h1>

<ul id="items"><li>First</li><li>Second</li></ul>

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

<input type="button" value="Add" onclick="addItem()">

<script>

function addItem(){

let inputElement = document.getElementById('newItemText');

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

li.textContent = inputElement.value;

let ul = document.getElementById('items');

ul.appendChild(li);

inputElement.value = '';

}

</script>

|  |
| --- |
| <h1>List of Items</h1> |
|  | <ul id="items"> |
|  | <li>First</li> |
|  | <li>Second</li> |
|  | </ul> |
|  | <input type="text" id="newItemText" /> |
|  | <input type="button" value="Add" onclick="addItem()"> |
|  | <script> |
|  | function addItem() { |
|  | let text = document.getElementById("newItemText").value; |
|  | let li = document.createElement("li"); |
|  | li.appendChild(document.createTextNode(text)); |
|  | document.getElementById("items").appendChild(li); |
|  | document.getElementById("newItemText").value = ""; |
|  | } |
|  | </script> |

1. **Add / Delete**

function addItem() {

let inputElement = document.getElementById('newText');

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

li.innerHTML = inputElement.value + ' <a href="#">[Delete]</a>';

li.lastChild.addEventListener('click', deleteItem)

let ul = document.getElementById('items');

ul.appendChild(li);

inputElement.value = '';

function deleteItem() {

// let li = this.parentNode;

// let ul = li.parentNode;

ul.removeChild(li);

}

}

<h1>List of Items</h1>

<ul id="items"></ul>

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

<input type="button" value="Add"

onclick="addItem()">

<script>

function addItem() {

let inputElement = document.getElementById('newText');

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

li.innerHTML = inputElement.value + ' <a href="#">[Delete]</a>';

li.lastChild.addEventListener('click', deleteItem)

let ul = document.getElementById('items');

ul.appendChild(li);

inputElement.value = '';

function deleteItem() {

// let li = this.parentNode;

// let ul = li.parentNode;

ul.removeChild(li);

}

}

</script>

|  |
| --- |
| <h1>List of Items</h1> |
|  | <ul id="items"></ul> |
|  | <input type="text" id="newText" /> |
|  | <input type="button" value="Add" onclick="addItem()"> |
|  | <script> |
|  | function addItem() { |
|  | let text = document.getElementById('newText').value; |
|  | let li = document.createElement("li"); |
|  | li.appendChild(document.createTextNode(text + " ")); |
|  | let span = document.createElement('span'); |
|  | span.innerHTML = "<a href='#'>[Delete]</a>"; |
|  | span.firstChild.addEventListener('click', deleteItem); |
|  | li.appendChild(span); |
|  | document.getElementById("items").appendChild(li); |
|  | document.getElementById('newText').value = ''; |
|  |  |
|  |  |
|  | function deleteItem() { |
|  | let li = this.parentNode.parentNode; |
|  | let ul = li.parentNode; |
|  | ul.removeChild(li); |
|  | } |
|  | } |
|  | </script> |

1. **Delete from Table**

function deleteByEmail() {

let emailToDelete = document.getElementsByName('email')[0].value;

let resultElement = document.getElementById('result');

let isDeleted = false;

let tdEmails = document.querySelectorAll('table tr td:nth-child(2)');

for (let tdEmail of tdEmails) {

if (tdEmail.textContent === emailToDelete) {

let trToDelete = tdEmail.parentNode;

trToDelete.parentNode.removeChild(trToDelete);

resultElement.textContent = 'Deleted.';

isDeleted = true;

}

}

if (!isDeleted) {

resultElement.textContent = 'Not found.';

}

}

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

<tr>

<th>Name</th>

<th>Email</th>

</tr>

<tr>

<td>Eve</td>

<td>eve@gmail.com</td>

</tr>

<tr>

<td>Nick</td>

<td>nick@yahooo.com</td>

</tr>

<tr>

<td>Didi</td>

<td>didi@didi.net</td>

</tr>

<tr>

<td>Tedy</td>

<td>tedy@tedy.com</td>

</tr>

</table>

Email: <input type="text" name="email" />

<button onclick="deleteByEmail()">Delete</button>

<div id="result" />

<script>

function deleteByEmail() {

let emailToDelete = document.getElementsByName('email')[0].value;

let resultElement = document.getElementById('result');

let isDeleted = false;

let tdEmails = document.querySelectorAll('table tr td:nth-child(2)');

for (let tdEmail of tdEmails) {

if (tdEmail.textContent === emailToDelete) {

let trToDelete = tdEmail.parentNode;

trToDelete.parentNode.removeChild(trToDelete);

resultElement.textContent = 'Deleted.';

isDeleted = true;

}

}

if (!isDeleted) {

resultElement.textContent = 'Not found.';

}

}

</script>

|  |
| --- |
| <table border="1" id="customers"> |
|  | <tr> |
|  | <th>Name</th> |
|  | <th>Email</th> |
|  | </tr> |
|  | <tr> |
|  | <td>Eve</td> |
|  | <td>eve@gmail.com</td> |
|  | </tr> |
|  | <tr> |
|  | <td>Nick</td> |
|  | <td>nick@yahooo.com</td> |
|  | </tr> |
|  | <tr> |
|  | <td>Didi</td> |
|  | <td>didi@didi.net</td> |
|  | </tr> |
|  | <tr> |
|  | <td>Tedy</td> |
|  | <td>tedy@tedy.com</td> |
|  | </tr> |
|  | </table> |
|  | Email: |
|  | <input type="text" name="email" /> |
|  | <button onclick="deleteByEmail()">Delete</button> |
|  | <div id="result" /> |
|  | <script> |
|  | function deleteByEmail() { |
|  | let emailToSearch = document.getElementsByName("email")[0].value; |
|  | let tds = document.querySelectorAll("#customers tr td:nth-child(2)"); |
|  | for (let td of tds) { |
|  | if (td.textContent == emailToSearch) { |
|  | let toDelete = td.parentNode; |
|  | toDelete.parentNode.removeChild(toDelete); |
|  | document.getElementById("result").textContent = "Deleted."; |
|  | return; |
|  | } |
|  | } |
|  | document.getElementById("result").textContent = "Not Found."; |
|  | } |
|  | </script> |

1. **Stopwatch**

function stopwatch(){

let startBtn = document.getElementById('startBtn');

let stopBtn = document.getElementById('stopBtn');

startBtn.addEventListener('click', startTimer);

stopBtn.addEventListener('click', stopTimer);

let time = 0;

let timeInterval;

function startTimer(){

//time = 0;

document.getElementById('time').textContent = '00:00';

timeInterval = setInterval(runTime, 1000);

startBtn.disabled = true;

stopBtn.disabled = false;

}

function stopTimer(){

clearInterval(timeInterval);

stopBtn.disabled = true;

startBtn.disabled = false;

}

function runTime(){

time++;

let seconds = time % 60;

if(seconds <= 9){

seconds = '0' + seconds;

}

let minutes = Math.trunc(time / 60);

if(minutes <= 9){

minutes = '0' + minutes;

}

document.getElementById('time').textContent = minutes + ':' + seconds;

}

}

<div id="time" style="border:3px solid blue; text-align:center; font-size:2em; margin-bottom:10px">00:00</div>

<button id="startBtn">Start</button>

<button id="stopBtn" disabled="true">Stop</button>

<script>

window.onload = function stopWatch() {

let startBtn = document.getElementById('startBtn');

let stopBtn = document.getElementById('stopBtn');

startBtn.addEventListener('click', startTimer);

stopBtn.addEventListener('click', stopTimer);

let time = 0;

let timeInterval;

function startTimer() {

time = 0;

document.getElementById('time').textContent = '00:00';

timeInterval = setInterval(runTime, 1000);

startBtn.disabled = true;

stopBtn.disabled = false;

}

function stopTimer() {

clearInterval(timeInterval);

stopBtn.disabled = true;

startBtn.disabled = false;

}

function runTime() {

time++;

let seconds = time % 60;

if (seconds <= 9) {

seconds = '0' + seconds;

}

let minutes = Math.trunc(time / 60);

if (minutes <= 9) {

minutes = '0' + minutes;

}

document.getElementById('time').textContent = minutes + ':' + seconds;

}

}

</script>

|  |
| --- |
| <div id="time" style="border:3px solid blue; text-align:center; font-size:2em; margin-bottom:10px">00:00</div> |
|  | <button id="startBtn">Start</button> |
|  | <button id="stopBtn" disabled="true">Stop</button> |
|  | <script> |
|  | window.onload = function () { |
|  | stopwatch(); |
|  |  |
|  | function stopwatch() { |
|  | let time, intervalID; |
|  | let startBtn = document.getElementById('startBtn'); |
|  | let stopBtn = document.getElementById('stopBtn'); |
|  | startBtn.addEventListener('click', function () { |
|  | time = -1; |
|  | incrementTime(); |
|  | intervalID = setInterval( |
|  | incrementTime, 1000); |
|  | startBtn.disabled = true; |
|  | stopBtn.disabled = false; |
|  | }); |
|  | stopBtn.addEventListener('click', function () { |
|  | clearInterval(intervalID); |
|  | startBtn.disabled = false; |
|  | stopBtn.disabled = true; |
|  | }); |
|  |  |
|  | function incrementTime() { |
|  | time++; |
|  | document.getElementById('time').textContent = |
|  | ("0" + Math.trunc(time / 60)).slice(-2) + |
|  | ':' + ("0" + (time % 60)).slice(-2); |
|  | } |
|  | } |
|  | } |
|  | </script> |

1. **Mouse Gradient**

function attachGradientEvents(){

let gradientElement = document.getElementById('gradient');

let resultElement = document.getElementById('result');

gradientElement.addEventListener('mousemove', showGradient);

gradientElement.addEventListener('mouseout', hideGradient);

function showGradient(event){

let gradient = event.offsetX / (event.target.clientWidth - 1);

gradient = Math.trunc(gradient \* 100);

resultElement.textContent = gradient + '%';

}

function hideGradient(){

resultElement.textContent = '';

}

}

<html>

<head>

<title>Mouse in Gradient</title>

<link rel="stylesheet" href="gradient.css" />

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

</head>

<body onload="attachGradientEvents()">

<div id="gradient-box">

<div id="gradient">Click me!</div>

</div>

<div id="result"></div>

</body>

<script>

function attachGradientEvents(){

let gradientElement = document.getElementById('gradient');

let resultElement = document.getElementById('result');

gradientElement.addEventListener('mousemove', showGradient);

gradientElement.addEventListener('mouseout', hideGradient);

function showGradient(event){

let gradient = event.offsetX / (event.target.clientWidth - 1);

gradient = Math.trunc(gradient \* 100);

resultElement.textContent = gradient + '%';

}

function hideGradient(){

resultElement.textContent = '';

}

}

</script>

<style>

#gradient-box {

width: 300px;

border: 2px solid lightgrey;

}

#gradient-box:hover {

border: 2px solid black;

}

#gradient {

height: 30px;

color: white;

text-shadow: 1px 1px 10px black;

text-align: center;

line-height: 30px;

background: linear-gradient(to right, black, white);

cursor: crosshair;

}

</style>

</html>

|  |
| --- |
| <html> |
|  |  |
|  | <head> |
|  | <title>Mouse in Gradient</title> |
|  | <link rel="stylesheet" href="gradient.css" /> |
|  | <script src="gradient.js"></script> |
|  | </head> |
|  |  |
|  | <body onload="attachGradientEvents()"> |
|  | <div id="gradient-box"> |
|  | <div id="gradient">Click me!</div> |
|  | </div> |
|  | <div id="result"></div> |
|  | </body> |
|  | <script> |
|  | function attachGradientEvents() { |
|  | let gradient = document.getElementById('gradient'); |
|  | gradient.addEventListener('mousemove', gradientMove); |
|  | gradient.addEventListener('mouseout', gradientOut); |
|  |  |
|  | function gradientMove(event) { |
|  | let power = event.offsetX / (event.target.clientWidth - 1); |
|  | power = Math.trunc(power \* 100); |
|  | document.getElementById('result').textContent = power + "%"; |
|  | } |
|  |  |
|  | function gradientOut(event) { |
|  | document.getElementById('result').textContent = ""; |
|  | } |
|  | }; |
|  | </script> |
|  | <style> |
|  | #gradient-box { |
|  | width: 300px; |
|  | border: 2px solid lightgrey; |
|  | } |
|  |  |
|  | #gradient-box:hover { |
|  | border: 2px solid black; |
|  | } |
|  |  |
|  | #gradient { |
|  | height: 30px; |
|  | color: white; |
|  | text-shadow: 1px 1px 10px black; |
|  | text-align: center; |
|  | line-height: 30px; |
|  | background: linear-gradient(to right, black, white); |
|  | cursor: crosshair; |
|  | } |
|  | </style> |
|  |  |
|  | </html> |

1. **Highlight Active**

function focus(){

let inputElements = document.getElementsByTagName('input');

Array.from(inputElements).forEach(input => {

input.addEventListener('focus', event => event.target.parentNode.className = 'focused');

input.addEventListener('blur', event => event.target.parentNode.removeAttribute('class'));

});

}

<!DOCTYPE html>

<html lang="en">

<head>

<meta charset="UTF-8">

<title>Focus</title>

<style>

div {

width: 470px;

}

div div {

text-align: center;

display: inline-block;

width: 200px;

height: 200px;

margin: 15px;

border: 1px solid #999;

}

.focused {

background: #999999;

}

</style>

</head>

<body onload="focus()">

<div>

<div>

<h1>Section 1</h1><input type="text" />

</div>

<div>

<h1>Section 2</h1><input type="text" />

</div>

<div>

<h1>Section 3</h1><input type="text" />

</div>

<div>

<h1>Section 4</h1><input type="text" />

</div>

</div>

<script>

function focus() {

let inputElements = document.getElementsByTagName('input');

Array.from(inputElements).forEach(input => {

input.addEventListener('focus', event => event.target.parentNode.className = 'focused');

input.addEventListener('blur', event => event.target.parentNode.removeAttribute('class'));

});

}

</script>

</body>

</html>

|  |
| --- |
| <!DOCTYPE html> |
|  | <html lang="en"> |
|  |  |
|  | <head> |
|  | <meta charset="UTF-8"> |
|  | <title>Focus</title> |
|  | <style> |
|  | div { |
|  | width: 470px; |
|  | } |
|  |  |
|  | div div { |
|  | text-align: center; |
|  | display: inline-block; |
|  | width: 200px; |
|  | height: 200px; |
|  | margin: 15px; |
|  | border: 1px solid #999; |
|  | } |
|  |  |
|  | .focused { |
|  | background: #999999; |
|  | } |
|  | </style> |
|  | </head> |
|  |  |
|  | <body onload="focus()"> |
|  | <div> |
|  | <div> |
|  | <h1>Section 1</h1> |
|  | <input type="text" /> |
|  | </div> |
|  | <div> |
|  | <h1>Section 2</h1> |
|  | <input type="text" /> |
|  | </div> |
|  | <div> |
|  | <h1>Section 3</h1> |
|  | <input type="text" /> |
|  | </div> |
|  | <div> |
|  | <h1>Section 4</h1> |
|  | <input type="text" /> |
|  | </div> |
|  | </div> |
|  | <script> |
|  | function focus() { |
|  | let inputs = document.getElementsByTagName('input'); |
|  | Array.from(inputs).forEach(i => { |
|  | i.addEventListener('focus', (event) => { |
|  | event.target.parentNode.className = 'focused'; |
|  | }); |
|  | i.addEventListener('blur', (event) => { |
|  | event.target.parentNode.removeAttribute('class'); |
|  | }); |
|  | }); |
|  | } |
|  | </script> |
|  | </body> |
|  |  |
|  | </html> |

1. **Dynamic Validation**

function validate(){

let inputEmail = document.getElementById('email');

inputEmail.addEventListener('change', validateEmail);

function validateEmail(){

let emailPattern = /^[a-z]+@[a-z]+\.[a-z]+$/;

let email = inputEmail.value;

if(emailPattern.test(email)){

inputEmail.removeAttribute('class');

}

else {

inputEmail.className = 'error';

}

}

}

<!DOCTYPE html>

<html lang="en">

<head>

<meta charset="UTF-8">

<title>Focus</title>

<style>

.error {

border: 2px solid red;

}

</style>

</head>

<body onload="validate()">

<label for="email">Enter email:</label>

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

<script>

function validate() {

let inputEmail = document.getElementById('email');

inputEmail.addEventListener('change', validateEmail);

function validateEmail() {

let emailPattern = /^[a-z]+@[a-z]+\.[a-z]+$/;

let email = inputEmail.value;

if (emailPattern.test(email)) {

inputEmail.removeAttribute('class');

}

else {

inputEmail.className = 'error';

}

}

}

</script>

</body>

</html>

|  |
| --- |
| <!DOCTYPE html> |
|  | <html lang="en"> |
|  |  |
|  | <head> |
|  | <meta charset="UTF-8"> |
|  | <title>Focus</title> |
|  | <style> |
|  | .error { |
|  | border: 2px solid red; |
|  | } |
|  | </style> |
|  | </head> |
|  |  |
|  | <body onload="validate()"> |
|  | <label for="email">Enter email:</label> |
|  | <input id="email" type="text" /> |
|  | <script> |
|  | function validate() { |
|  | document.querySelector('input') |
|  | .addEventListener('change', onChange); |
|  | let regex = /^([\w\-.]+)@([a-z]+)(**\.**[a-z]+)+$/; |
|  |  |
|  | function onChange(event) { |
|  | if (!regex.test(event.target.value)) |
|  | event.target.className = 'error'; |
|  | else |
|  | event.target.removeAttribute('class'); |
|  | } |
|  | } |
|  | </script> |
|  | </body> |
|  |  |
|  | </html> |