

MIT WORLD PEACE UNIVERSITY

Full Stack Development  
Third Year B. Tech, Semester 5

---

---

IMPLEMENTATION OF FORM VALIDATION,  
JQUERY, AND AJAX

---

---

LAB ASSIGNMENT 3

Prepared By

Krishnaraj Thadesar  
Cyber Security and Forensics  
Batch A1, PA 20

September 21, 2023

# Contents

<b>1 Aim</b>	<b>1</b>
<b>2 Objectives</b>	<b>1</b>
<b>3 Problem Statement</b>	<b>1</b>
<b>4 Theory</b>	<b>1</b>
<b>5 Types of Form Validation</b>	<b>1</b>
5.1 Types . . . . .	1
<b>6 The Document Object Model</b>	<b>2</b>
6.1 DOM Tree . . . . .	2
6.2 History . . . . .	3
6.3 Advantages . . . . .	3
<b>7 JQuery</b>	<b>4</b>
7.1 History and Significance . . . . .	4
7.2 Advantages . . . . .	4
7.3 JQuery Selectors . . . . .	5
<b>8 Platform</b>	<b>7</b>
<b>9 Input and Output</b>	<b>7</b>
<b>10 Screenshots</b>	<b>8</b>
<b>11 Code</b>	<b>12</b>
<b>12 Conclusion</b>	<b>18</b>
<b>13 FAQ</b>	<b>19</b>

## 1 Aim

Write a program to perform following form validations using JavaScript, where all fields are mandatory, and some fields like Phone number, Email Address, Zip code Validation etc. are included. Also include JavaScript to access and manipulate Document Object Model (DOM) objects in an HTML web page. Include JQuery to develop to develop your application as an Ajax based application.

## 2 Objectives

- To understand what is form validation.
- To learn basic functioning of DOM objects.
- To learn how to apply various techniques to implement it.

## 3 Problem Statement

Write a program to design Student registration form by using HTML, CSS having following fields: Username, Email, Phone number, Password, Confirm Password and write external javascript code to achieve following validations:

1. Fields should not be empty. If spaces are entered those should be considered empty
2. Phone number must accept only numeric values and it should be 10 digits
3. Password length must be at least 7 and it should contain at least one capital letter, one digit and one special character from the set (&,\$,#, )
4. Value entered in password field and confirm password fields must match

Email address must contain @ sign and a ., there should be few letters before the @ sign, should be three letters between @ sign and a . There must be 3 or 2 letters after the . (hint: Use regular expression)

## 4 Theory

## 5 Types of Form Validation

**Definition 1** *Form validation is the process of checking that a form has been filled in correctly before it is submitted. It is used to prevent users from submitting invalid data, such as an email address without an @ sign or a password that is too short.*

### 5.1 Types

1. **Client-side validation:** Client-side validation is performed by the web browser before the form is submitted to the server. It is used to provide immediate feedback to the user about any errors in their input.
2. **Server-side validation:** Server-side validation is performed by the web server after the form has been submitted. It is used to ensure that the data submitted by the user is valid and safe to process.

3. **Database validation:** Database validation is performed by the database management system after the data has been stored in the database. It is used to ensure that the data is consistent with the database schema and any constraints defined on the table.
4. **Business logic validation:** Business logic validation is performed by the application after the data has been retrieved from the database. It is used to ensure that the data is consistent with the business rules defined by the application.

## 6 The Document Object Model

**Definition 2** *The Document Object Model (DOM) is a programming interface for HTML and XML documents. It defines the logical structure of a document and the way it is accessed and manipulated.*

1. The DOM is used by JavaScript to access and manipulate HTML documents.
2. It is also used by other programming languages, such as PHP and Python, to generate HTML documents dynamically.
3. The DOM is an object-oriented model that represents HTML documents as a collection of objects.
4. Each object has properties and methods that can be used to access and manipulate the document.
5. The DOM is a platform-independent model that can be used with any programming language.

### 6.1 DOM Tree

**Definition 3** *The DOM tree is a hierarchical representation of the HTML document. It consists of nodes that represent elements, attributes, text, and comments.*

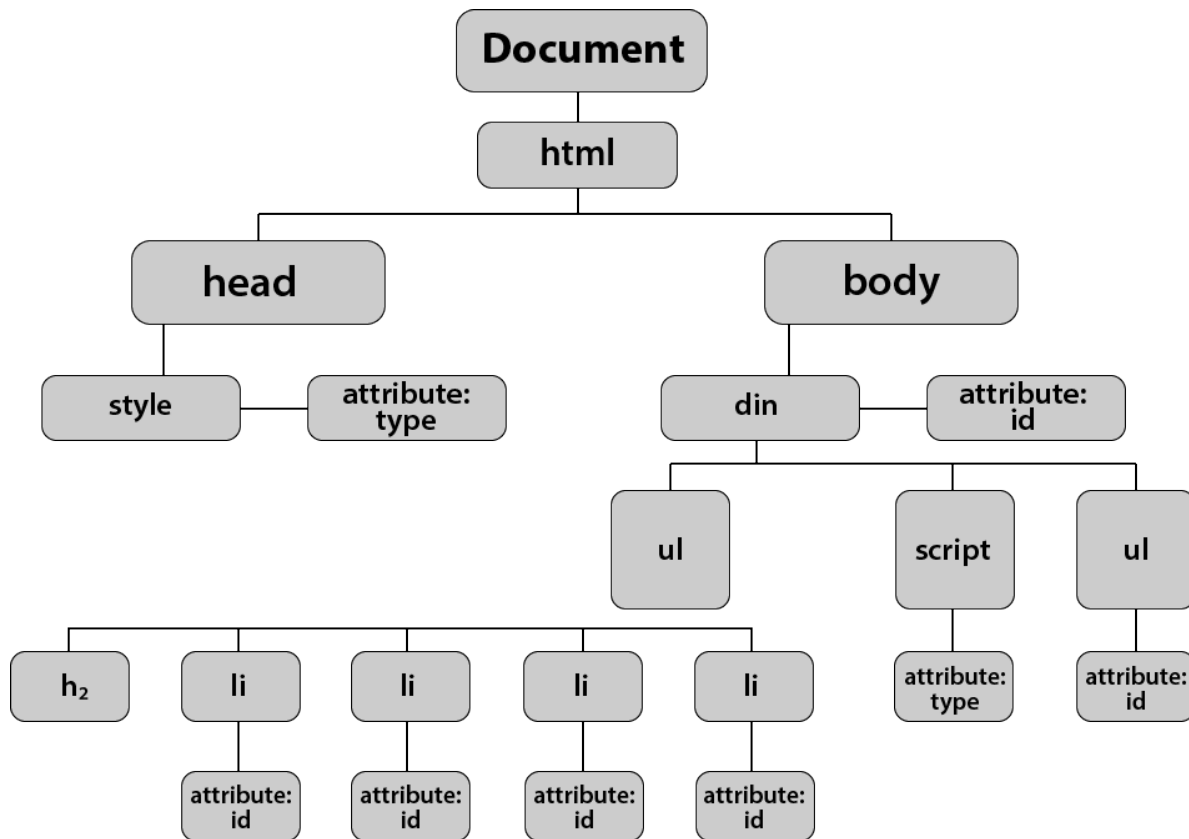


Figure 1: The DOM Tree

## 6.2 History

- The DOM was first introduced in 1998 by the World Wide Web Consortium (W3C).
- It was designed to be a platform-independent model for representing HTML documents.
- It is now supported by all major web browsers, including Internet Explorer, Firefox, Chrome, Safari, and Opera.

## 6.3 Advantages

1. The DOM is a platform-independent model that can be used with any programming language.
2. It is a standard model that is supported by all major web browsers.
3. It is an object-oriented model that represents HTML documents as a collection of objects.
4. It is a hierarchical model that represents the logical structure of an HTML document.
5. It is a dynamic model that can be used to access and manipulate HTML documents.
6. It is a flexible model that can be extended to support new features.
7. It is a powerful model that can be used to create interactive web applications.
8. It is a simple model that is easy to learn and understand.

9. It is a scalable model that can be used to create large web applications.

## 7 JQuery

**Definition 4** *JQuery is a JavaScript library that simplifies the process of creating interactive web pages. It is used to manipulate HTML documents and handle events.*

### 7.1 History and Significance

1. JQuery was first released in 2006 by John Resig.
2. It was designed to simplify the process of creating interactive web pages.
3. It is now supported by all major web browsers, including Internet Explorer, Firefox, Chrome, Safari, and Opera.
4. It is used by many popular websites, such as Google, Facebook, Twitter, and YouTube.
5. It is used by many popular web applications, such as WordPress, Drupal, Joomla, and Magento.

### 7.2 Advantages

1. **Simplified DOM Manipulation:** jQuery simplifies DOM (Document Object Model) manipulation tasks. It provides easy-to-use methods to select, modify, and manipulate HTML elements.

```
1 <!-- HTML -->
2 <button id="myButton">Click Me</button>
3
4 <!-- jQuery -->
5 <script src="https://code.jquery.com/jquery-3.6.0.min.js"></script>
6 <script>
7     // Selecting an element by ID
8     var button = $("#myButton");
9
10    // Modifying its text
11    button.text("New Text");
12 </script>
```

Listing 1: Example: Selecting and Modifying Elements

2. **Event Handling:** jQuery simplifies event handling by providing methods like ‘.on()’ and ‘.click()’ for attaching event handlers to elements.

```
1 <!-- HTML -->
2 <button id="myButton">Click Me</button>
3
4 <!-- jQuery -->
5 <script src="https://code.jquery.com/jquery-3.6.0.min.js"></script>
6 <script>
7     // Event handling with jQuery
8     $("#myButton").click(function() {
9         alert("Button Clicked!");
10    });
11 </script>
```

Listing 2: Example: Event Handling

3. **AJAX Support:** jQuery streamlines AJAX (Asynchronous JavaScript and XML) requests, making it easier to fetch data from a server without reloading the entire page.

```
1 // jQuery AJAX request
2 $.ajax({
3   url: "https://api.example.com/data",
4   method: "GET",
5   success: function(data) {
6     console.log(data);
7   },
8   error: function(error) {
9     console.error(error);
10  }
11 });
```

Listing 3: Example: AJAX Request

4. **Cross-Browser Compatibility:** jQuery abstracts away browser-specific differences, ensuring consistent behavior across various browsers.
5. **Rich Set of Plugins:** jQuery offers a vast ecosystem of plugins that extend its functionality, allowing developers to add features like sliders, date pickers, and more without reinventing the wheel.
6. **Animation and Effects:** jQuery provides methods for creating animations and adding visual effects to web pages with ease.

```
1 <!-- HTML -->
2 <div id="box">Click Me</div>
3
4 <!-- jQuery -->
5 <script src="https://code.jquery.com/jquery-3.6.0.min.js"></script>
6 <script>
7   // jQuery animation
8   $("#box").click(function() {
9     $(this).animate({ left: "200px" }, "slow");
10  });
11 </script>
```

Listing 4: Example: Animation

### 7.3 JQuery Selectors

jQuery provides a variety of selectors that allow you to target and manipulate specific elements in the HTML DOM. Here are some commonly used jQuery selectors:

1. **Element Selector:** Selects all elements with a specified tag name.

```
1 <!-- HTML -->
2 <ul>
3   <li>Item 1</li>
4   <li>Item 2</li>
5   <li>Item 3</li>
6 </ul>
7
8 <!-- jQuery -->
9 <script src="https://code.jquery.com/jquery-3.6.0.min.js"></script>
```

```
10 <script>
11     // Select all <li> elements
12     var listItems = $("li");
13 </script>
```

Listing 5: Example: Element Selector

2. **ID Selector:** Selects a single element with a specific ID attribute.

```
1 <!-- HTML -->
2 <div id="myDiv">Hello, World!</div>
3
4 <!-- jQuery -->
5 <script src="https://code.jquery.com/jquery-3.6.0.min.js"></script>
6 <script>
7     // Select the element with ID "myDiv"
8     var myElement = $("#myDiv");
9 </script>
```

Listing 6: Example: ID Selector

3. **Class Selector:** Selects all elements with a specific class attribute.

```
1 <!-- HTML -->
2 <p class="highlight">This is a highlighted text.</p>
3 <p>This is a normal text.</p>
4
5 <!-- jQuery -->
6 <script src="https://code.jquery.com/jquery-3.6.0.min.js"></script>
7 <script>
8     // Select all elements with class "highlight"
9     var highlightedElements = $(".highlight");
10 </script>
```

Listing 7: Example: Class Selector

4. **Attribute Selector:** Selects elements with a specific attribute and optional value.

```
1 <!-- HTML -->
2 <input type="text" id="username" name="username" value="John" />
3
4 <!-- jQuery -->
5 <script src="https://code.jquery.com/jquery-3.6.0.min.js"></script>
6 <script>
7     // Select element with attribute "name"
8     var usernameInput = $("[name='username']");
9 </script>
```

Listing 8: Example: Attribute Selector

5. **Descendant Selector:** Selects all elements that are descendants of a specified element.

```
1 <!-- HTML -->
2 <ul id="myList">
3     <li>Item 1</li>
4     <li>Item 2</li>
5 </ul>
6
7 <!-- jQuery -->
```



```
8 <script src="https://code.jquery.com/jquery-3.6.0.min.js"></script>
9 <script>
10     // Select all <li> elements inside the element with ID "myList"
11     var listItems = $("#myList li");
12 </script>
```

Listing 9: Example: Descendant Selector

## 8 Platform

**Operating System:** Arch Linux x86-64

**IDEs or Text Editors Used:** Visual Studio Code

**Compilers or Interpreters:** Brave Browser (Chromium v117.0.5938.88.)

## 9 Input and Output

1. The Student Validation form was created, and the relevant fields were added.
2. The validation was done on client side using Javascript Successfully.
3. The screenshots of the same are given below.

## 10 Screenshots

The screenshot shows a 'Student Registration Form' on a light pink background. The form is a white rounded rectangle with the title 'Student Registration Form' at the top. Below the title is a prompt: 'Please enter your details below to register for the course.' The form contains several input fields with red validation messages below them:

- Name:** A text input field with the placeholder 'Enter your name'. Below it is the message 'Please enter a valid name.'
- Parents Name:** A text input field with the placeholder 'Enter your parents name'. Below it is the message 'Please enter a valid parents name.'
- Email:** A text input field with the placeholder 'Enter your email'. Below it is the message 'Please enter a valid email address.'
- Country Code:** A dropdown menu with 'Select a country' and a downward arrow. Below it is the message 'Please select a country code.'
- Phone Number:** A text input field with the placeholder 'Enter your phone number'. Below it is the message 'Please enter a valid phone number.'
- Date of Birth:** A date input field with the placeholder 'mm/dd/yyyy' and a calendar icon. Below it is the message 'Please enter a valid date of birth.'

At the bottom of the form is a blue 'Submit' button.

Figure 2: The Student Validation form with Mandatory Fields. You cant submit without filling them. The Phone Number field only accepts numbers.

## Student Registration Form

Please enter your details below to register for the course.

DEU

GHA

GIB

GRC

GRL

GRD

GLP

GUM

GTM

GGY

GIN

GNB

GUY

HTI

HMD

HND

HKG

HUN

ISL

IND

IND

thadesar

ame.

ress.

Phone Number

9834312135

Please select a country code. Please enter a valid phone number.

Date of Birth

mm/dd/yyyy

Please enter a valid date of birth.

Submit

Figure 3: The Country Code Selection Field. The Countries REST api is called by using JQuery AJAX to get the list of countries.


## Student Registration Form

Please enter your details below to register for the course.


**Name**

**Parents Name**

**Email**

 Please enter a part followed by '@'. '@wer.com' is incomplete.

**Date of Birth**



Please enter a valid date of birth.

Figure 4: Email Validation done on client side JS

The image shows a web form for course registration. At the top, a dark grey toast notification displays the message "127.0.0.1:5500 says Form submitted successfully!" with an "OK" button. Below this, the form instructs the user to "Please enter your details below to register for the course." The form fields include: "Name" (Krishnaraj Prashant Thadesar), "Parents Name" (Prashant Thadesar), "Email" (something@something.com), "Country Code" (IND), "Phone Number" (9834312135), and "Date of Birth" (09/15/2023). A red error message "Please enter a valid date of birth." is visible below the date field. A blue "Submit" button is at the bottom of the form.

127.0.0.1:5500 says  
Form submitted successfully!

OK

Please enter your details below to register for the course.

**Name**  
Krishnaraj Prashant Thadesar

**Parents Name**  
Prashant Thadesar

**Email**  
something@something.com

**Country Code** **Phone Number**  
IND 9834312135

**Date of Birth**  
09/15/2023

Please enter a valid date of birth.

Submit

Figure 5: Form being submitted after entering everything. Notice the button is blue.

127.0.0.1:5500 says  
Form submitted successfully!

OK

Please enter your details below to register for the course.

**Name**  
Krishnaraj Prashant Thadesar

**Parents Name**  
Prashant Thadesar

**Email**  
something@something.com

**Country Code** **Phone Number**  
IND 9834312135

**Date of Birth**  
09/15/2023

Submit

Figure 6: The Color of the button changes to green for 3 seconds, as instructed in the problem statement. Only then the form submits.

## 11 Code

```
1 <!DOCTYPE html>
2 <html lang="en">
3   <head>
4     <meta charset="UTF-8" />
5     <title>Student Registration Form</title>
```

```
6     <link
7         rel="stylesheet"
8         href="https://cdnjs.cloudflare.com/ajax/libs/tailwindcss/2.2.7/tailwind.min.
9         css"
10    />
11 </head>
12 <body class="bg-pink-100">
13     <div class="mx-auto my-10 bg-white rounded-3xl shadow-sm w-1/3 p-10">
14         <h2 class="font-medium mb-10 text-4xl self-center text-center mt-5">
15             Student Registration Form
16         </h2>
17         <div class="text-xl text-center my-10">
18             Please enter your details below to register for the course.
19         </div>
20         <form>
21             <div class="mb-4">
22                 <label
23                     class="block text-gray-700 font-bold mb-2"
24                     for="name"
25                 >
26                     Name
27                 </label>
28                 <input
29                     class="border rounded-md py-2 px-3 text-gray-700 leading-tight focus:
30                     outline-none focus:shadow-outline"
31                     id="name"
32                     type="text"
33                     placeholder="Enter your name"
34                 />
35                 <p class="text-red-500 text-xs mt-1 hidden" id="name_error">
36                     Please enter a valid name.
37                 </p>
38             </div>
39             <div class="mb-4">
40                 <label
41                     class="block text-gray-700 font-bold mb-2"
42                     for="parents_name"
43                 >
44                     Parents Name
45                 </label>
46                 <input
47                     class="border rounded-md py-2 px-3 text-gray-700 leading-tight focus:
48                     outline-none focus:shadow-outline"
49                     id="parents_name"
50                     type="text"
51                     placeholder="Enter your parents name"
52                 />
53                 <p
54                     class="text-red-500 text-xs mt-1 hidden"
55                     id="parents_name_error"
56                 >
57                     Please enter a valid parents name.
58                 </p>
59             </div>
60             <div class="mb-4">
61                 <label
62                     class="block text-gray-700 font-bold mb-2"
63                     for="email"
64                 >
```

```
62     Email
63     </label>
64     <input
65         class="border rounded-md py-2 px-3 text-gray-700 leading-tight focus:
outline-none focus:shadow-outline"
66         id="email"
67         type="email"
68         placeholder="Enter your email"
69     />
70     <p
71         class="text-red-500 text-xs mt-1 hidden"
72         id="email_error"
73     >
74         Please enter a valid email address.
75     </p>
76 </div>
77 <div class="mb-4 flex">
78     <div class="mr-2">
79         <label
80             class="block text-gray-700 font-bold mb-2"
81             for="phone_country_code"
82         >
83             Country Code
84         </label>
85         <select
86             class="border rounded-md py-2 px-3 text-gray-700 leading-tight focus
:outline-none focus:shadow-outline"
87             id="phone_country_code"
88         >
89             <option value="">Select a country</option>
90         </select>
91         <p
92             class="text-red-500 text-xs mt-1 hidden"
93             id="phone_country_code_error"
94         >
95             Please select a country code.
96         </p>
97     </div>
98     <div class="flex-1">
99         <label
100             class="block text-gray-700 font-bold mb-2"
101             for="phone_number"
102         >
103             Phone Number
104         </label>
105         <input
106             class="border rounded-md py-2 px-3 text-gray-700 leading-tight focus
:outline-none focus:shadow-outline"
107             id="phone_number"
108             type="tel"
109             placeholder="Enter your phone number"
110         />
111         <p
112             class="text-red-500 text-xs mt-1 hidden"
113             id="phone_number_error"
114         >
115             Please enter a valid phone number.
116         </p>
117     </div>
```



```
118     </div>
119     <div class="mb-4">
120         <label
121             class="block text-gray-700 font-bold mb-2"
122             for="date_of_birth"
123         >
124             Date of Birth
125         </label>
126         <input
127             class="border rounded-md py-2 px-3 text-gray-700 leading-tight focus:
outline-none focus:shadow-outline"
128             id="date_of_birth"
129             type="date"
130         />
131         <p
132             class="text-red-500 text-xs mt-1 hidden"
133             id="date_of_birth_error"
134         >
135             Please enter a valid date of birth.
136         </p>
137     </div>
138     <div class="mt-6">
139         <button
140             class="bg-blue-500 hover:bg-blue-700 text-white font-bold py-2 px-4
rounded focus:outline-none focus:shadow-outline"
141             type="submit"
142         >
143             Submit
144         </button>
145     </div>
146 </form>
147 </div>
148 <script src="https://code.jquery.com/jquery-3.6.0.min.js"></script>
149 <script>
150     $(document).ready(function () {
151         const nameInput = $("#name");
152         const parentsNameInput = $("#parents_name");
153         const emailInput = $("#email");
154         const phoneCountryCodeInput = $("#phone_country_code");
155         const phoneNumbersInput = $("#phone_number");
156         const dateOfBirthInput = $("#date_of_birth");
157         const nameError = $("#name_error");
158         const parentsNameError = $("#parents_name_error");
159         const emailError = $("#email_error");
160         const phoneNumbersError = $("#phone_number_error");
161         const dateOfBirthError = $("#date_of_birth_error");
162         const phoneCountryCodeError = $("#phone_country_code_error");
163
164         function validateName() {
165             if (nameInput.val().trim() === "") {
166                 nameError.removeClass("hidden");
167                 return false;
168             } else {
169                 nameError.addClass("hidden");
170                 return true;
171             }
172         }
173
174         function validateParentsName() {
```

```
175         if (parentsNameInput.val().trim() === "") {
176             parentsNameError.removeClass("hidden");
177             return false;
178         } else {
179             parentsNameError.addClass("hidden");
180             return true;
181         }
182     }
183
184     function validateEmail() {
185         const emailRegex = /^[^\s@]+@[^\s@]+\.[^\s@]+$/;
186         if (!emailRegex.test(emailInput.val())) {
187             emailError.removeClass("hidden");
188             return false;
189         } else {
190             emailError.addClass("hidden");
191             return true;
192         }
193     }
194
195     function validatePhoneNumber() {
196         const phoneRegex = /^[0-9]{10}$/;
197         if (!phoneRegex.test(phoneNumberInput.val())) {
198             phoneNumberError.removeClass("hidden");
199             return false;
200         } else {
201             phoneNumberError.addClass("hidden");
202             return true;
203         }
204     }
205
206     function validateDateOfBirth() {
207         const dobRegex = /^\d{4}-\d{2}-\d{2}$/;
208         if (!dobRegex.test(dateOfBirthInput.val())) {
209             dateOfBirthError.removeClass("hidden");
210             return false;
211         } else {
212             dateOfBirthError.addClass("hidden");
213             return true;
214         }
215     }
216
217     function validatePhoneCountryCode() {
218         if (phoneCountryCodeInput.val() === "") {
219             phoneCountryCodeError.removeClass("hidden");
220             return false;
221         } else {
222             phoneCountryCodeError.addClass("hidden");
223             return true;
224         }
225     }
226
227     function validateForm() {
228         const isNameValid = validateName();
229         const isParentsNameValid = validateParentsName();
230         const isEmailValid = validateEmail();
231         const isPhoneNumberValid = validatePhoneNumber();
232         const isDateOfBirthValid = validateDateOfBirth();
233         const isPhoneCountryCodeValid = validatePhoneCountryCode();
```

```
234         return (
235             isNameValid &&
236             isParentsNameValid &&
237             isEmailValid &&
238             isPhoneNumberValid &&
239             isDateOfBirthValid &&
240             isPhoneCountryCodeValid
241         );
242     }
243
244     $.ajax({
245         url: "https://restcountries.com/v3.1/all",
246         type: "GET",
247         success: async function (data) {
248             // Sort data by country name in ascending order
249             await data.sort((a, b) =>
250                 a.name.common.localeCompare(b.name.common)
251             );
252             const countrySelect = $("#phone_country_code");
253             $.each(data, function (index, country) {
254                 const option = $("
```

```
293         }, 3000);  
294     }  
295     });  
296     });  
297     });  
298     </script>  
299     </body>  
300 </html>
```

Listing 10: "index.html"

## 12 Conclusion

Thus, we have successfully implemented form validation using Javascript and used JQuery as well as JQuery Ajax.

## 13 FAQ

### 1. Reasons for Form Validations:

- **Data Integrity:** Form validations ensure that the data submitted by users is accurate and follows the expected format. This prevents incorrect or malicious data from being processed, maintaining the integrity of your application's database.
- **Enhanced User Experience:** Validations provide immediate feedback to users, helping them correct errors before submission. This enhances the user experience and reduces frustration.
- **Security:** Proper form validations can prevent common security vulnerabilities like SQL injection and cross-site scripting (XSS) attacks by rejecting potentially harmful input.

### 2. Modifying Attribute Value using DOM: In JavaScript, you can modify an attribute value using the DOM (Document Object Model) as follows:

```
1 // Assuming 'element' is a reference to the HTML element you want to modify.
2
3 // To change the 'src' attribute of an image:
4 element.setAttribute('src', 'new_image.jpg');
5
6 // To modify the 'href' attribute of a link:
7 element.setAttribute('href', 'new_url.html');
8
```

Listing 11: Example: Modifying Attribute Value using DOM

### 3. jQuery Ajax: jQuery Ajax is a feature in the jQuery library that allows you to make asynchronous HTTP requests to a server and handle the responses without requiring a full page reload. It offers a simplified way to perform tasks like fetching data from a server, sending data to a server, or updating a web page without the need for extensive JavaScript code.

```
1 // jQuery AJAX request
2 $.ajax({
3   url: "https://api.example.com/data",
4   method: "GET",
5   success: function(data) {
6     console.log(data);
7   },
8   error: function(error){
9     console.error(error);
10  }
11 });
12
```

Listing 12: Example: AJAX Request