Ex No: 3 **VALIDATION OF WEBFORM USING HTML**

04-09-2021

AIM

To implement Client side Scripts for Validating Web Form Controls using DHTML.

ALGORITHM

- The form will include one text field called "Your Name", and a submit button.
- Validation script will ensure that the user enters their name before the form is sent to the server.
- Open this page to see it in action.
- Try pressing the **Send Details** button without filling anything in the "Your Name" field.
- You might like to open the source code for this form in a separate window
- The page consists of a JavaScript function called validate_form() that performs the form validation, followed by the form itself.

CONCEPTS INVOLVED:

Dynamic HTML, or DHTML, is a collection of technologies used together to create interactive and animated websites by using a combination of a static markup language (such as HTML), a client-side scripting language (such as JavaScript), a presentation definition language (such as CSS), and the Document Object Model (DOM). The application of DHTML was introduced by Microsoft with the release of Internet Explorer 4 in 1997.

DHTML allows scripting languages to change variables in a web page's definition language, which in turn affects the look and function of otherwise "static" HTML page content, after the page has been fully loaded and during the viewing process. Thus the dynamic characteristic of DHTML is the way it functions while a page is viewed, not in its ability to generate a unique page with each page load.

By contrast, a dynamic web page is a broader concept, covering any web page generated differently for each user, load occurrence, or specific variable values. This includes pages created by client-side scripting, and ones created by server-side scripting (such as PHP, Python, JSP or ASP.NET) where the web server generates content before sending it to the client.

DHTML is differentiated from Ajax by the fact that a DHTML page is still request/reload-based. With DHTML, there may not be any interaction between the client and server after the page is loaded; all processing happens in JavaScript on the client side. By contrast, an Ajax page uses features of DHTML to initiate a request (or 'subrequest') to the server to perform additional actions. For example, if there are multiple tabs on a page, pure DHTML approach would load the contents of all tabs and then dynamically display only the one that is active, while AJAX could load each tab only when it is really needed.

A webform, web form or HTML form on a web page allows a user to enter data that is sent to a server for processing. Forms can resemble paper or database forms because web users fill out the forms using checkboxes, radio buttons, or text fields. For example, forms can be used to enter shipping or credit card data to order a product, or can be used to retrieve search results from a search engine.

Forms can be made up of standard graphical user interface elements:

- <text> — a simple text box that allows input of a single line of text.
- <email> - a type of <text> that requires a partially validated email address
- <number> - a type of <text> that requires a number
- <password> — similar to <text>, it is used for security purposes, in which the characters typed in are invisible or replaced by symbols such as *
- <radio> — a radio button
- <file> — a file select control for uploading a file
- <reset> — a reset button that, when activated, tells the browser to restore the values to their initial values.
- <submit> — a button that tells the browser to take action on the form (typically to send it to a server)
- <textarea> — much like the <text> input field except a <textarea> allows for multiple rows of data to be shown and entered
- <select> — a drop-down list that displays a list of items a user can select from

Program

INDEX.HTML

```
<html>
```

```
<body>
<section class="s2">
<div class="main-container">
<a href=""></a>
<h3 style="text-align: center;">Get In Touch</h3>
<form name="welcome message" method='POST' netlify-honeypot="bot-field" data
netlify="true" id="contact-form">
 <a name="contact"></a>
 <label>*Name</label>
 <input id="first" class="inputfield" type="text" name="name" required>

 <label>Subject</label>
 <input id="second" class="inputfield" type="text" name="subject">

<label>*Email</label>
 <input id="third" class="inputfield" type="email" name="email" required>

 <label>*Phone</label>
 <input id="third" class="inputfield" type="tel" id="phone" name="phone" pattern="[0-9]{3}-
[0-9]{2}-[0-
9]{3}"
 name="email" required>
 <small style="color: white;">Note: Enter a valid Indian phone
number</small><br><br>

 <label>*Message</label>
 <textarea id="fourth" class="inputfield" name="message" required></textarea>
 <button onclick="clicked()" id="submitbtn" type="submit" value="Send">Send</button>
 </form>
</div>

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

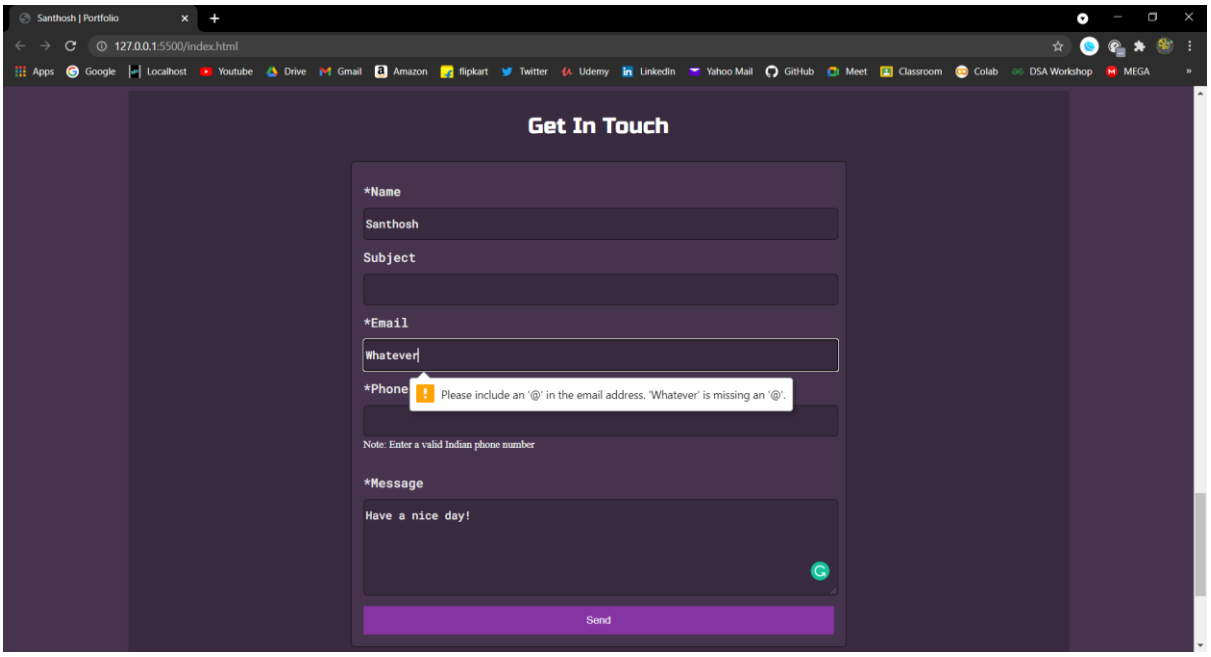
INDEX.JS

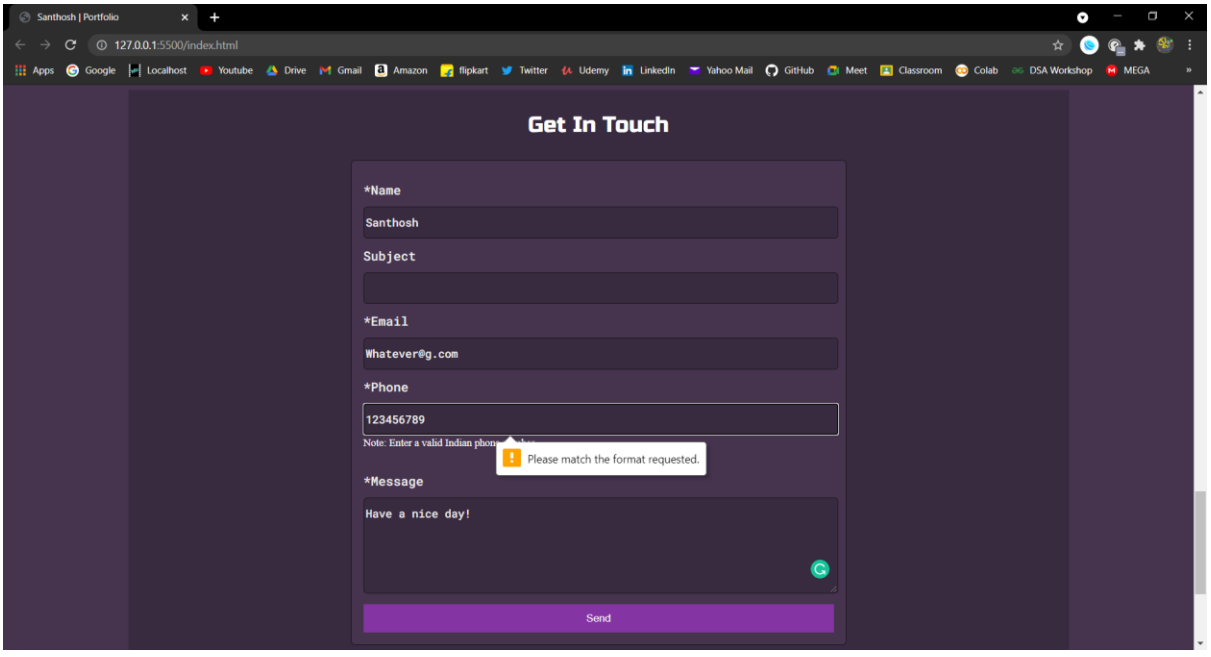
```
function clicked(){
var a = document.getElementById('first').value;

var b = document.getElementById('second').value;

var c = document.getElementById('third').value;

var d = document.getElementById('fourth').value;

 if ((a && c && d) =="") {
alert("Please Enter all required information and press send!");
}
}
```

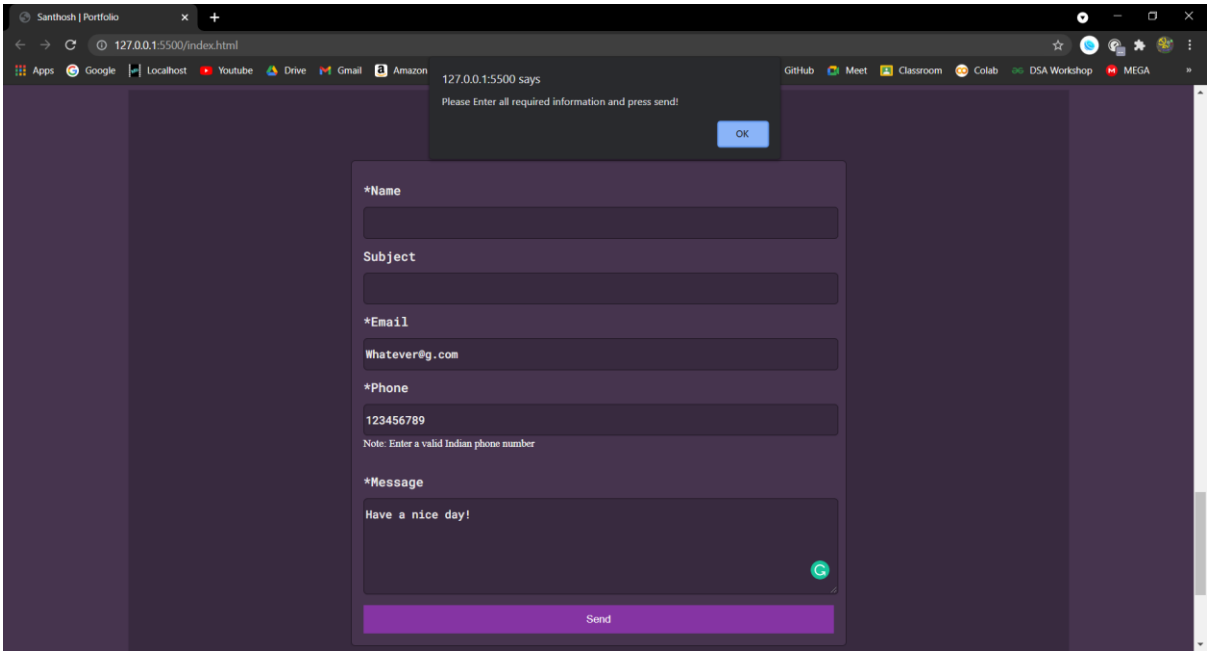
STYLING:

BOOTSTRAP USED AS STYLING FRAMEWORK

OUTPUT:


Santhosh | Portfolio
127.0.0.1:5500/index.html
Get In Touch
*Name
Santhosh
Subject
*Email
Whatever
*Phone
Please include an '@' in the email address. 'Whatever' is missing an '@'.
Note: Enter a valid Indian phone number
*Message
Have a nice day!
Send


Santhosh | Portfolio
127.0.0.1:5500/index.html
Get In Touch
*Name
Santhosh
Subject
*Email
Whatever@g.com
*Phone
123456789
Please match the format requested.
*Message
Have a nice day!
Send



RESULT

In this experiment, Client Side Scripts for validating web form controls using DHTML has been implemented.