



Lab Exercise 05– Design and Validate HTML Form.

Objectives

By the end of this lab session you will be able to:

- Familiar with form design and form validation.

Prerequisites

Computer literacy, a reasonable understanding of HTML, CSS, and JavaScript.

Introduction

HTML Forms

HTML Forms are one of the main points of **interaction** between a **user** and a **website**. They allow users to send data to the website. To create HTML form, you need to use **<form>** tag.



```
<html>

<head>

    <title>My First Form</title>
</head>
<body>

    <form>

    </form>

</body>

</html>|
```

Lab Exercise 05– Design and Validate HTML Form.

HTML Form Attributes

Example 1: Below shows form attributes.

```
<html>

<head>

    </head>
<body>

    <form name="myForm" action="contactDetails.html" target="_blank" method="GET">

    </form>
</body>

</html>
```



Attributes	Description
name	Specifies the name of a form.
action	Defines the action to be performed when the form is submitted.
target	Specifies if the submitted result will open in a new browser tab, a frame, or in the current window. When you specify the target as _blank linked document (contactDetails.html) page will open in a new window or tab
method	specifies the HTTP method (GET or POST) to be used when submitting the form data

Lab Exercise 05– Design and Validate HTML Form.

Form Elements

Between `<form>` and `</form>` tags you can use **form elements** like **text fields**, **checkboxes**, **radio buttons**, **submit buttons**, and more.

```
<html>

<head>

    <title>My First Form</title>
</head>
<body>

    <form>
        // form elements
        <input type="text"/>
    </form>
</body>

</html>|
```



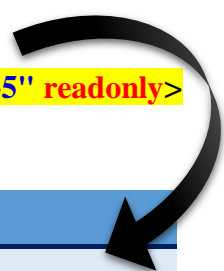
Type	Description
<code><input type="text"></code>	Defines a one-line text input field
<code><input type="radio"></code>	Defines a radio button (for selecting one of many choices)
<code><input type="submit"></code>	Defines a submit button (for submitting the form)

Lab Exercise 05– Design and Validate HTML Form.

HTML Input Attributes

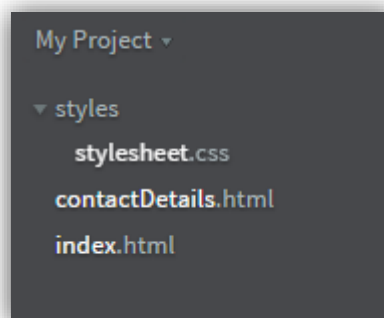
Input field contains attributes like name, type, value and more.

<input type="text" name="FirstName" value="john" maxlength="10" size="35" readonly>



Attribute	Description
name	Specifies the name of the <input> element
value	specifies the initial value for an input field
readonly	specifies that the input field is read only (cannot be changed)
disabled	specifies that the input field is disabled
size	specifies the size (in characters) for the input field
maxlength	specifies the maximum allowed length for the input field

Exercise 1: You need to create a folder called **My Project** and add files called **index.html**, **contactDetails.html** and **stylesheet.css** like below.





Lab Exercise 05– Design and Validate HTML Form.

Then write below code in the **index.html**

```
<html>

<head>
  <title>My First Form</title>
  <link rel="stylesheet" type="text/css" href="styles/stylesheet.css"/>
</head>
<body>

  <h2>User Details Form</h2>
  <form name="myForm" action="contactDetails.html" target="_blank" method="GET">
    <label>First name:</label><br>
    <input type="text" name="firstname" value="">
    <br>
    <label>Last name:</label><br>
    <input type="text" name="lastname" value="">
    <br>
    <label>Gender:</label><br>
    <input type="radio" name="gender" value="male"><span>Male</span>
    <input type="radio" name="gender" value="female"><span>Female</span><br>

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

  </form>
</body>

</html>
```

Write below code in the **contactDetails.html**

```
<html>

<head>

  <title>Contact Details</title>
</head>
<body>

  <h1>This is contact details page</h1>

</body>

</html>
```



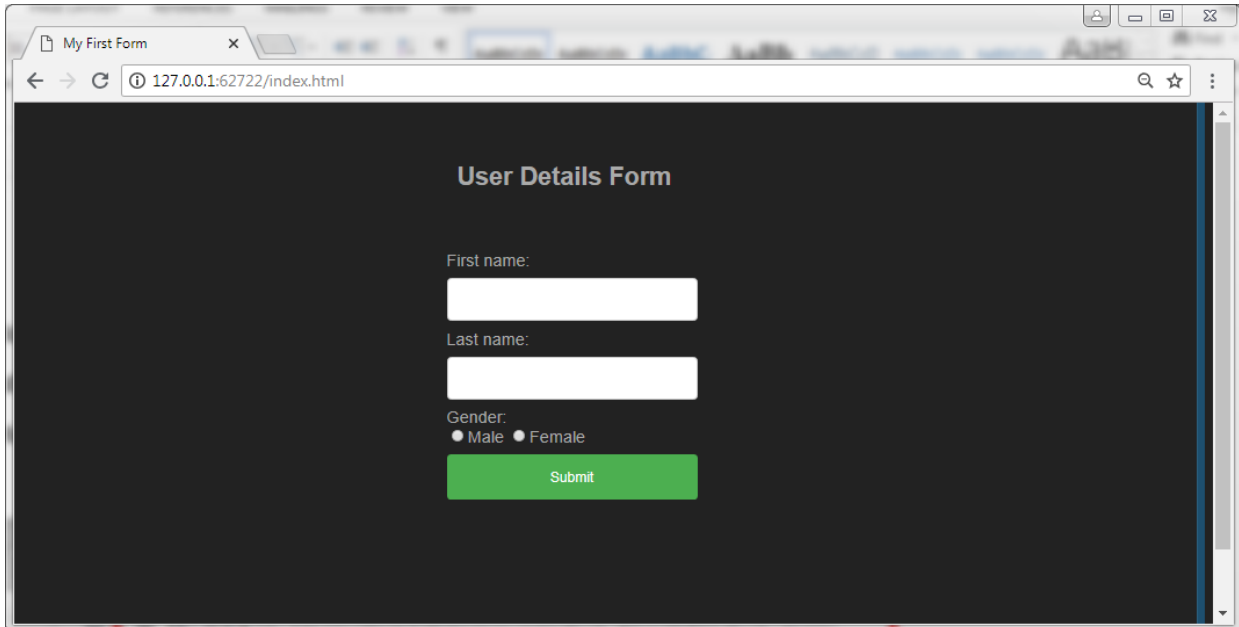
Lab Exercise 05– Design and Validate HTML Form.

To add style to your first form use below styles in your **stylesheet.css**

```
body, html{
    height: 100%;
    background: #222222;
    font-family: 'Lato', sans-serif;
}
label, p, span{
    color: #AAAAAA;
}
h2{
    color: #AAAAAA;
    width: 300px;
    margin: 5% auto;
}
input[type=text], select {
    width: 75%;
    padding: 12px 20px;
    margin: 8px 0;
    display: inline-block;
    border: 1px solid #ccc;
    border-radius: 4px;
    box-sizing: border-box;
}
input[type=submit] {
    width: 75%;
    background-color: #4CAF50;
    color: white;
    padding: 14px 20px;
    margin: 8px 0;
    border: none;
    border-radius: 4px;
    cursor: pointer;
}
form{
    text-align: left;
    width: 320px;
    margin: 0 auto;
}
```

Lab Exercise 05– Design and Validate HTML Form.

When you open **index.html** in the browser your output will be like below.

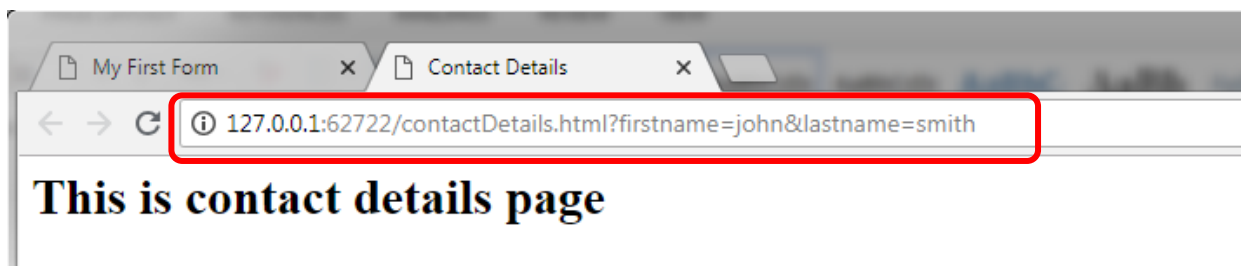


The screenshot shows a web browser window with the title 'My First Form'. The address bar displays '127.0.0.1:62722/index.html'. The main content area has a dark background and is titled 'User Details Form'. It contains the following fields and controls:

- 'First name:' label followed by a white text input field.
- 'Last name:' label followed by a white text input field.
- 'Gender:' label followed by two radio buttons: 'Male' and 'Female'.
- A green 'Submit' button at the bottom.

Enter your details and click on the **submit** button, then you will navigate to a new page called contact details.

Note: When you use **GET** method you will see all the inputted form details in the URL.





Lab Exercise 05– Design and Validate HTML Form.

Form Validation

Different types of form validation

There are two different types of form validation.

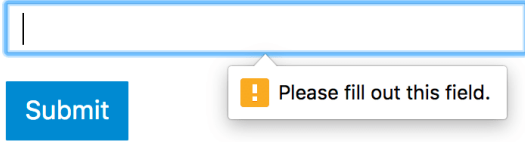
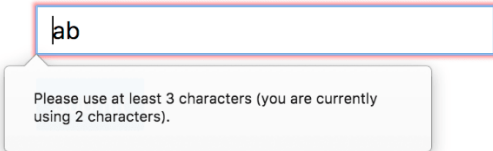
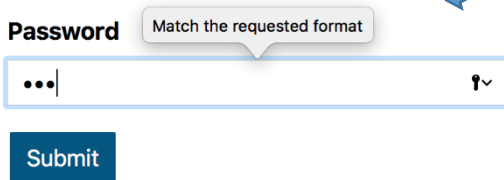
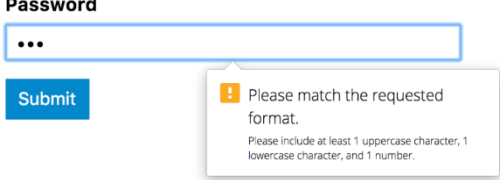
- **Client-side validation**
 - It occurs in the browser, before the data has been submitted to the server.
 - This is more user-friendly than server-side validation as it gives an instant response.
 - **This can be further subdivided:**
 - **JavaScript validation**
 - Coded using JavaScript.
 - It is completely customizable.
 - **Built-in form validation**
 - HTML5 form validation features.
 - This generally does not require JavaScript.
 - Built-in form validation has better performance, but it is not as customizable as JavaScript.
- **Server-side validation** is validation which occurs on the server, after the data has been submitted.

In the real world, developers tend to use a **combination** of **client-side** and **server-side validation**.

Lab Exercise 05– Design and Validate HTML Form.

Basic Text Validation

Through a combination of semantic input types (for example, `<input type="email">`) and validation attributes (such as `required` and `pattern`), browsers can natively validate form inputs and alert users when they're doing it wrong.

Validation attributes	Description
required	<p>Alert users who don't fill it out and refuse to let them submit the form.</p> <pre><input type="text" required></pre> <p>Text Input</p> 
minlength and maxlength	<p>Requires a value to be between 3 and 12 characters in length.</p> <pre><input type="text" minlength="3" maxlength="12"></pre> <p>Text Input</p> 
pattern	<p>Required passwords to contain at least 1 uppercase character, 1 lowercase character, and 1 number, the browser can validate that for you.</p> <pre><input type="password" pattern="^(?=.*\d)(?=.*[a-z])(?=.*[A-Z])(?!.*\s).*\$" required></pre> <div>   </div>
title	<p>The title value will be included with any error message if the pattern doesn't match.</p> <pre><input type="password" pattern="^(?=.*\d)(?=.*[a-z])(?=.*[A-Z])(?!.*\s).*\$" title="Please include at least 1 uppercase character, 1 lowercase character, and 1 number." required></pre>

Lab Exercise 05– Design and Validate HTML Form.

Exercise 2: Complete below exercise and understand javascript validation.

```
<html>
  <head>
    <title> Form Validation:Demo Preview </title>
    <meta name="robots" content="noindex, nofollow">

    <link rel="stylesheet" type="text/css" href="formvalid.css">
    <script src="formvalid.js"></script>

  </head>
  <body>

    <div id="main">

      <div id="form_layout">
        <h2>JavaScript Form Validation</h2>

        <!-- Form starts from here-->
        <form onsubmit='return formValidation()'><br><p id="head"></p><br>
          <label>Full Name:</label>
          <input type='text' id='firstname' /><br />
          <p id="p1"></p> <!--This segment displays the validation rule for name-->

          <label>Username (6-8 characters):</label>
          <input type='text' id='username' /><br />
          <p id="p2"></p> <!--This segment displays the validation rule for user name-->

          <label>Email:</label>
          <input type='text' id='email' /><br />
          <p id="p3"></p> <!--This segment displays the validation rule for email-->

          <label>State:</label>
          <select id='state'>
            <option>Please Choose</option>
            <option>America</option>
            <option>Australia</option>
            <option>Sweden</option>
            <option>Africa</option>
          </select><br />
          <p id="p4"></p> <!--This segment displays the validation rule for selection-->

          <label>Address:</label>
          <input type='text' id='addr' /><br />
          <p id="p5"></p> <!--This segment displays the validation rule for address-->

          <label>Zip Code: </label>
          <input type='text' id='zip' /><br />
          <p id="p6"></p> <!--This segment displays the validation rule for zip-->

          <input type='submit' id="submit" value='Check Form' /><br />
        </form>
      </div>
    </div>
  </body>
</html>
```

Formvalid.html



Lab Exercise 05– Design and Validate HTML Form.

```
function formValidation(){

    // Make quick references to our fields
    var firstname = document.getElementById('firstname');
    var addr = document.getElementById('addr');
    var zip = document.getElementById('zip');
    var state = document.getElementById('state');
    var username = document.getElementById('username');
    var email = document.getElementById('email');

    // to check empty form fields.

    if(firstname.value.length == 0){
        document.getElementById('head').innerText = "* All fields are mandatory *";
        //this segment displays the validation rule for all fields
        firstname.focus();
        return false;
    }

    // Check each input in the order that it appears in the form!
    if(inputAlphabet(firstname, "* For your name please use alphabets only *")){

        if(lengthDefine(username, 6, 8)){

            if(emailValidation(email, "* Please enter a valid email address *")){

                if(trueSelection(state, "* Please Choose any one option")){

                    if(textAlphanumeric(addr, "* For Address please use numbers and
letters *")){

                        if(textNumeric(zip, "* Please enter a valid zip code *")){

                            return true;
                        }
                    }
                }
            }
        }
    }

    return false;
}

//function that checks whether input text is numeric or not.
function textNumeric(inputtext, alertMsg){
    var numericExpression = /^[0-9]+$/;
    if(inputtext.value.match(numericExpression)){
        return true;
    }else{

```

formvalid.js



Lab Exercise 05– Design and Validate HTML Form.

```

        document.getElementById('p6').innerText = alertMsg; //this segment
displays the validation rule for zip
        inputtext.focus();
        return false;
    }
}

//function that checks whether input text is an alphabetic character or not.
function inputAlphabet(inputtext, alertMsg){
    var alphaExp = /^[a-zA-Z]+$/;
    if(inputtext.value.match(alphaExp)){
        return true;
    }else{
        document.getElementById('p1').innerText = alertMsg; //this segment
displays the validation rule for name
        //alert(alertMsg);
        inputtext.focus();
        return false;
    }
}

//function that checks whether input text includes alphabetic and numeric
characters.
function textAlphanumeric(inputtext, alertMsg){
    var alphaExp = /^[0-9a-zA-Z]+$/;
    if(inputtext.value.match(alphaExp)){
        return true;
    }else{
        document.getElementById('p5').innerText = alertMsg; //this segment
displays the validation rule for address
        inputtext.focus();
        return false;
    }
}

// Function that checks whether the input characters are restricted according
to defined by user.
function lengthDefine(inputtext, min, max){
    var uInput = inputtext.value;
    if(uInput.length >= min && uInput.length <= max){
        return true;
    }else{
        document.getElementById('p2').innerText = "* Please enter between "
+min+ " and " +max+ " characters *"; //this segment displays the validation
rule for username
        inputtext.focus();
        return false;
    }
}

```



Lab Exercise 05– Design and Validate HTML Form.

```
// Function that checks whether a option is selected from the selector and if
it's not it displays an alert message.

function trueSelection(inputtext, alertMsg){
    if(inputtext.value == "Please Choose"){
        document.getElementById('p4').innerText = alertMsg; //this segment
displays the validation rule for selection
        inputtext.focus();
        return false;
    }else{
        return true;
    }
}

// Function that checks whether an user entered valid email address or not
and displays alert message on wrong email address format.

function emailValidation(inputtext, alertMsg){
    var emailExp = /^[\\w\\-\\.\\+]+@[a-zA-Z0-9\\.\\-]+\\. [a-zA-z0-9]{2,4}$/;
    if(inputtext.value.match(emailExp)){
        return true;
    }else{
        document.getElementById('p3').innerText = alertMsg; //this segment
displays the validation rule for email
        inputtext.focus();
        return false;
    }
}
```



Lab Exercise 05– Design and Validate HTML Form.

```
/*-----  
css settings for HTML div exactCenter  
-----*/  
@import url(http://fonts.googleapis.com/css?family=Raleway);  
  
#main{  
width:960px;  
margin:50px auto;  
font-family:raleway;  
}  
  
span{  
color:red;  
}  
  
h2{  
background-color: #FEFFED;  
text-align:center;  
border-radius: 10px 10px 0 0;  
margin: -10px -40px;  
padding: 30px;  
}  
  
hr{  
border:0;  
border-bottom:1px solid #ccc;  
margin: 10px -40px;  
margin-bottom: 30px;  
}  
  
#form_layout{  
width:300px;  
float: left;  
border-radius: 10px;  
font-family:raleway;  
border: 2px solid #ccc;  
padding: 10px 40px 25px;  
margin-top: -2px;  
}  
  
input[type=text],input[type=password],#state{  
width:99.5%;  
padding: 10px;  
margin-top: 8px;  
border: 1px solid #ccc;  
padding-left: 5px;  
font-size: 16px;  
font-family:raleway;  
}  
  
input[type=submit]{  
width: 100%;  
background-color:#FFBC00;  
color: white;  
border: 2px solid #FFCB00;  
padding: 10px;  
font-size:20px;
```

formvalid.css



Lab Exercise 05– Design and Validate HTML Form.

```
cursor:pointer;
border-radius: 5px;
margin-bottom: 15px;
}

a{
text-decoration:none;
color: cornflowerblue;
}

i{
color: cornflowerblue;
}

p{
    font-size:12px;
    color:red;
}

/*-----
css settings for right side advertisement
-----*/
#formget{
float:right;
}
```

You need to take below output.



Lab Exercise 05– Design and Validate HTML Form.

Form Validation: Demo Pr x

127.0.0.1:61250/formvalid.html

JavaScript Form Validation

Full Name:

Username(6-8 characters):

Email:

State:

Please Choose ▼

Address:

Zip Code:

Check Form