**Module 3: - Introduction and Learning Objectives**

HTML5 has many elements that enable developers to create well-structured and varied websites. These HTML5 elements provide ways to separate an HTML document into divisions, create headers and footers, define sections, create headings, and delineate the body of the document. In addition, there are many HTML5 elements that allow the user to interact with the website, inputting information in various formats like dates, times, numbers, email addresses, and much more besides.  To be able to create the structure you want for your site, you must be familiar with some of these elements.

After completing this module, you will be able to:

* Describe the basic structural elements of HTML5.
* Describe the attributes of HTML5 input elements.

# HTML5 Tags and Structural Elements

Hello. Welcome to HTML5 Tags and Structural Elements.

After watching this video, you will be able to:

List specific elements to HTML5 and

Describe HTML5 structural elements:

section, article, header, footer, figure, figcaption.

There are many HTML5 specific elements. The table shows some of them.

Notice that the HTML5 element tags have intuitive names, such as audio that is used to embed

sound content, and canvas that is used for graphical content.

Not all of these element tags are covered in this video, so take a moment to read through

the table.

This table shows more HTML5 elements.

The last tag in the table is the comments tag.

Comments in HTML can assist the developer to explain what the code does.

The browsers do not display these comments, unless the viewer views the source.

In a division-based layout, each part of a page that you want to format differently is

separated into its own division.

Think of the natural divisions into which you can divide the page that is shown here.

These divisions might include the page title, a banner with selectable tabs, an area for

the image, and further blocks for text and thumbnail images.

The <div> tag was not introduced in HTML5, but is a good starting point for laying out

a page or document.

Then, you can use other HTML5-specific structural elements inside the <div> tag.

Most HTML elements have a semantic meaning. In other words, the element describes either the function or the type of data that is contained within.

For example, a <p> element is a paragraph that usually contains some text.

In contrast, the <div> tag has no innate semantic meaning besides the logical grouping of the content.

You can use these dedicated elements to mark up your website, rather than using the generic

<div> or <span> elements.

An <article> defines a block of code that can be distinguished from the rest of the

page.

A <section> defines a logical separation in the document such as the numbered chapters

of an online manual.

A <header> element is used to group the headers for a page.

The <footer> defines the area at the bottom of the page.

This code example shows some of the HTML5 structural element.

Here, the <article> element is used to create an online news report.

The <article> includes a header.

Within the <article>, the page is marked up into sections that have their own headers.

The page also includes a <footer> element.

The <aside> tag is used to provide additional information that is related to the main discussion.

The aside tag lets you extract and display further content or go to additional resources

without detracting from the main discussion.

The <figure> tag defines a self-contained element that is referred to from the main

content.

The <figure> element can be used to embed graphics, images, or code sections.

The <figurecaption> tag defines the caption for the contents of the <figure> element.

Websites typically have sections that are dedicated to navigational links that go to

these sites or for moving between pages of a single application.

In HTML5 syntax, these navigational links are placed inside a <nav> tag.

The <nav> tag is a convenience tag for grouping navigational links in a web page.

Websites typically have sections that are dedicated to navigational links that go to

these sites or for moving between pages of a single application.

In HTML5 syntax, these navigational links are placed inside a <nav> tag.

The <nav> tag is a convenience tag for grouping navigational links in a web page.

In this video, you learned:

Tags provide control within an HTML5 document.

Some tags provide structural elements:

The <div> tag separates areas in a document into divisions, enabling you to apply different

styles to different parts of a document.

Dedicated elements like <article>, <section>, <header>, and <footer> are more specific than

the generic <div> element.

The <aside>, <figure> and <figcaption> tags enable you to group content.

And the <nav> tags enable you to group navigational links.

# HTML5 Input Element: Attributes for the Input Tag

Hello. Welcome to HTML5 Input Element: Attributes for the Input Tag.

After watching this video, you will be able to:

Describe the attributes of the HTML5 input element input type attributes tel, email,

date time, number, range, and color.

The < input type= "color" / > attribute allows the user to select a color.

The dialog varies depending on the browser.

Some browsers don’t support this input type.

In non-supporting browsers, this input type is displayed as a regular text input field

into which the user might type a valid color name or color code.

The < input type="date" > attribute is a date control (year, month, day) with no time zone.

The example shows the input type="date" field as it is seen in Google Chrome.

The input dialog varies from browser to browser.

The datetime-local attribute provides input for a date and time (year, month, day, hour,

minute, AM/PM) with no time zone.

The datetime-local input field is displayed as a drop-down calendar in the Google Chrome

browser. The time can be typed or entered using the spinner control.

The < input type="email" > attribute is displayed as a regular text input field.

It provides feedback when the input does not follow the email format.

The < input type="number" > takes a numeric value as input.

You can optionally specify the minimum, maximum values, step size, etc.

The example shows the number selector in the Chrome browser. Only the numbers between 5

and the 10 are available for selection.

The < input type="range" > takes a numeric range as input.

The example shows the number selector in the Google Chrome browser.

Only the numbers in the range of the minimum and the maximum are available for selection.

The range attribute displays a slider with the range of values between the minimum and

maximum.

Only the slider itself is shown. Additional JavaScript code is needed in order to display

the value of the slider.

The differences between < input type="search"\ > and < input type="text" / > are mostly in

style.

WebKit-based browsers return a history of recently searched text strings.

The search input field on the Safari browser has rounded corners.

The input type="tel" pattern="[parameters]" attribute expects a telephone number as input.

On its own, the < input type="tel" > provides nothing more than a text entry field in the

browsers.

It does not enforce numeric only input since many telephone numbers include other characters,

such as the plus sign and hyphens.

You need to supply your own pattern matcher if you want the browser to validate the telephone

number.

The URL attribute is used to validate that the user typed in a properly formatted URL

or web address.

The < input list="some\_list" > uses the <datalist> feature.

Not to be confused with the < select > element. The <datalist> options are only suggestions.

Useful for auto-complete functionality.

You can fill the list by nesting <option> elements inside the <datalist> tag.

These options are the types of fruits listed in the drop-down list.

Placeholder text is used to provide hints of what the input text format looks like.

The placeholder fills the input text field with the example values in a lighter shade

of text.

The form does not submit the placeholder text value if the input text is not overwritten.

The required attribute implies that some text must be typed.

The requirement to type some input applies even if the field contains placeholder text.

What happens if browser-based validation is not supported for these input attributes?

There are several options to performing validation in browsers that do not support all HTML5

input attributes.

You can use JavaScript and JQuery libraries.

You can assume that more browsers will support these features over time, and leave all final

validation to server-side processing.

You can code client-side validation that is attached to the form submit event handler

to validate all the fields on the form when the form is submitted.

In this video, you learned:

Common attributes of the < input > tag.

How to implement validation fallback.

# Additional HTML Elements: HTML fieldset and legend Tag

Logo

Description automatically generated

## **Special HTML Elements**

In this section, we will explore a tag found within the <form> tag called **<fieldset>** tag and **<legend>** tag.

**Duration:** 15 min

## **Objectives**

After completing this reading section you will be able to:

1. Use **<fieldset>** Tag appropriately in HTML pages
2. Use **<legend>** Tag to decorate your fieldset.

### **HTML fieldset tag**

* The HTML **<fieldset>** tag is found within the **<form>** tag and is used to group related elements in an HTML form inside a box.
* There is no restriction to the kind of elements that can be inside a fieldset. But they are mostly used to group related input type of elements, as shown in the example below.
* The elements grouped in a fieldset can be . This element can be specially useful in large forms, where readability and ease of access can be improved with segmentation. Browsers will most likely render a frame around the grouped controls.

### **Syntax**

<fieldset> Contents... </fieldset>

### **Attribute**

1. **disabled**: It specifies that the elements belonging to the fieldset should be disabled.
2. **form**: It specifies id of the form the fieldset is to be considered a part of.
3. **name**: It specifies the name for the fieldset.

### **Example**

In the first example we will try to create a form:

<!DOCTYPE **html**>

<**html**>

<**body**>

<**h1**>The fieldset element</**h1**>

<**form**>

<**fieldset** form="user\_regn" name="user\_details">

<**label** for="fname">First name:</**label**>

<**input** type="text" id="fname" name="fname"><**br**><**br**>

<**label** for="lname">Last name:</**label**>

<**input** type="text" id="lname" name="lname"><**br**><**br**>

<**label** for="email">Email:</**label**>

<**input** type="email" id="email" name="email"><**br**><**br**>

<**label** for="birthday">Birthday:</**label**>

<**input** type="date" id="birthday" name="birthday"><**br**><**br**>

</**fieldset**>

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

</**form**>

</**body**>

</**html**>

### **Output**

Graphical user interface, application

Description automatically generated

**Note: You can edit the form as per your convenient**

### **HTML legend tag**

A **fieldset** can additionally have a title or name, that can be provided by **legend**. The **<legend>** tag is used with the **<fieldset>** element as a first child to define the caption for the grouped related fields This tag is also commonly referred to as the **<fieldset>** element. By using **<legend>** tag with **<fieldset>** elements, it is easy to understand the purpose of grouped form elements.

### **Example**

To understand the **<legend>** tag, let's add this tag to the above example and see what will be the output:

<!DOCTYPE **html**>

<**html**>

<**body**>

<**h1**>The fieldset element</**h1**>

<**form**>

<**fieldset** form="user\_regn" name="user\_details"> <**legend**>Personal Details:</**legend**>

<**label** for="fname">First name:</**label**>

<**input** type="text" id="fname" name="fname"><**br**><**br**>

<**label** for="lname">Last name:</**label**>

<**input** type="text" id="lname" name="lname"><**br**><**br**>

<**label** for="email">Email:</**label**>

<**input** type="email" id="email" name="email"><**br**><**br**>

<**label** for="birthday">Birthday:</**label**>

<**input** type="date" id="birthday" name="birthday"><**br**><**br**>

</**fieldset**>

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

</**form**>

</**body**>

</**html**>

### **Output**

Graphical user interface, application

Description automatically generated

### **Congratulations! You have completed the exercise.**

## **Author(s)**

#### **Sourabh Mohan**

## **Changelog**

| **Date** | **Version** | **Changed by** | **Change Description** |
| --- | --- | --- | --- |
| 2021-08-31 | 1. | Sourabh | Initial version created based |

# Hands-On Lab: HTML5

**Step 1 of 1**

# Lab: Create an HTML form

**Estimated time**: 30 minutes

An event management company is looking for part-time employees and wants interested candidates to register online. They plan to contact those who register whenever there is an event. The company has approached you to create an online form that the applicants can use to register.

## Objectives

After completing this lab, you will be able to:

1. **Create** an HTML **form**

2. **Add** labels **and** text **input** **fields**

3. **Add** a control **to** **input** a phone number

4. **Add** a control **to** **input** an email **ID**

5. **Add** a \*\*Submit\*\* button

6. **Group** the **form** elements **in** a fieldset **and** **add** a legend **to** it.

## 1. Create a form

In the window to the right, click **File > New File**. A **New File** window opens.

Type registration\_form.html as the file name and click **OK**. You are now ready to start creating your web page.

You will use the <form> tag to create an HTML form. The action attribute of the form specifies the URL that processes the data when the form is submitted.

**Note**: The code at the destination URL is written by the back-end system developer. As a front-end developer, your focus is on creating the HTML form.

The following code creates a basic form within an HTML document, to which you will add controls. Copy the code into your HTML file and save it.

<!DOCTYPE **html**>

<**html**>

<**head**>

<**title**> Registration Form </**title**>

</**head**>

<**body**>

<**h2**>Registration Form</**h2**>

<**form** action="/register">

</**form**>

</**body**>

</**html**>

## 2. Add a label and a text input field

The <input> tag is used to create different types of controls that enable the user to input values. The default input type is text.

The <label> tag defines the label text for other HTML form elements. The for attribute of the <label> tag should be the same as the ID of the element it binds to.

Let's add a label and a text input field for the first name of the user. Type the following code inside the <form> tag.

<label **for**="firstname">First name :</label>

<input type="text" id="firstname" name="firstname">

The id attribute enables the front-end developer to access this input control. The name attribute enables the back-end system developer to access the value that's entered in this input field.

We need a similar field for the user's last name:

<label **for**="lastname">Last name :</label>

<input type="text" id="lastname" name="lastname">

After you add the two input text fields, your code should look like this:

<!DOCTYPE **html**>

<**html**>

<**head**>

<**title**> Registration Form </**title**>

</**head**>

<**body**>

<**h2**>Registration Form</**h2**>

<**form** action="/register">

<**label** for="firstname">First name :</**label**>

<**input** type="text" id="firstname" name="firstname">

<**br**><**br**>

<**label** for="lastname">Last name :</**label**>

<**input** type="text" id="lastname" name="lastname">

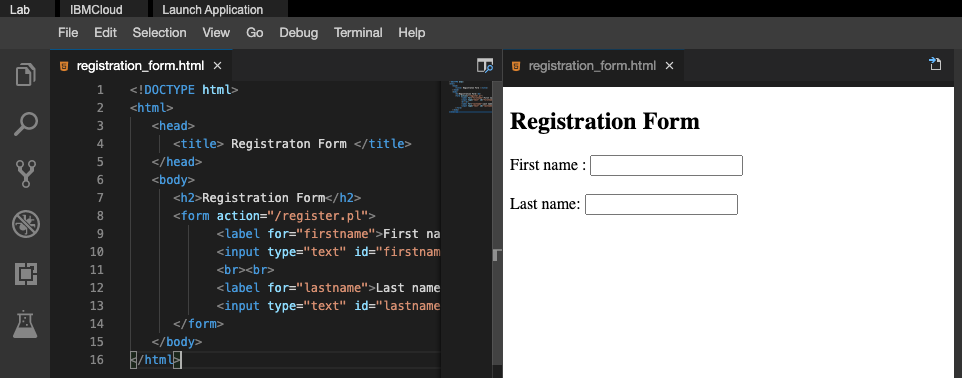
</**form**>

</**body**>

</**html**>

Note that the <br> tag creates a line break. Without the <br> tag, all our form controls would display in a single line in our HTML page.

To see how your HTML page displays, click the preview icon at the top right of the window. Your page should now look like this:



## 3. Add a control to input an email ID

To enable users to enter their e-mail address, you will use the <input> tag with the email type. Add the following code to your form after the **Last name** text field:

<label **for**="email">Email :</label>

<input type="email" id="email" name="email">

Your page should now like this:

<!DOCTYPE **html**>

<**html**>

<**head**>

<**title**> Registration Form </**title**>

</**head**>

<**body**>

<**h2**>Registration Form</**h2**>

<**form** action="/register">

<**label** for="firstname">First name :</**label**>

<**input** type="text" id="firstname" name="firstname">

<**br**><**br**>

<**label** for="lastname">Last name :</**label**>

<**input** type="text" id="lastname" name="lastname">

<**br**><**br**>

<**label** for="email">Email :</**label**>

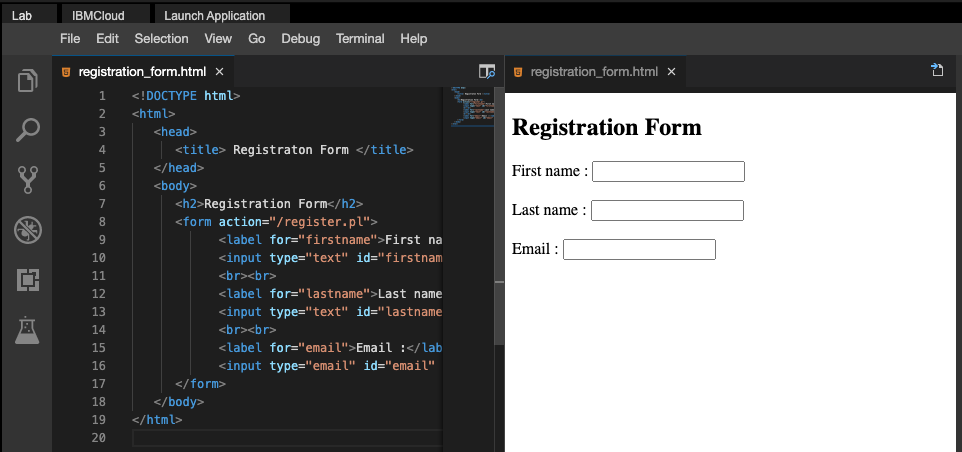
<**input** type="email" id="email" name="email">

</**form**>

</**body**>

</**html**>

When you have updated your code, save the file. The page should now look like this:



## Add a Submit button

Finally, let's add a **Submit** button at the end of the form. When a user clicks **Submit**, the form data is submitted to the back-end program for processing.

The following code will create a **Submit** button with a text label. Add this line immediately after the **Email** field:

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

Your final form code should look like this:

<!DOCTYPE **html**>

<**html**>

<**head**>

<**title**> Registration Form </**title**>

</**head**>

<**body**>

<**h2**>Registration Form</**h2**>

<**form** action="/register">

<**label** for="firstname">First name :</**label**>

<**input** type="text" id="firstname" name="firstname">

<**br**><**br**>

<**label** for="lastname">Last name :</**label**>

<**input** type="text" id="lastname" name="lastname">

<**br**><**br**>

<**label** for="email">Email :</**label**>

<**input** type="email" id="email" name="email">

<**br**><**br**>

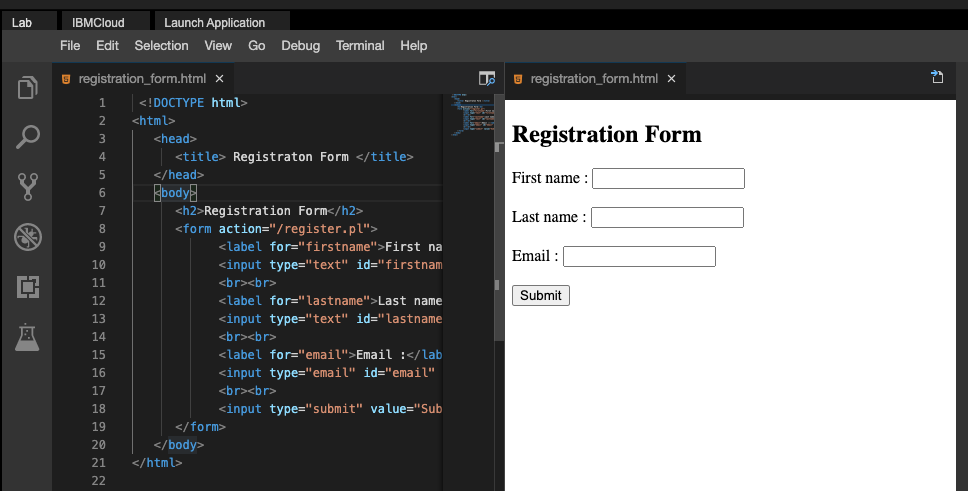
<**input** type="submit" value="Submit">

</**form**>

</**body**>

</**html**>

When you have updated the code, save the file. The page should now look like this:



## Add a fieldset

Now that the form is complete. We can use the **fieldset** tag to group the related elements of the form. The following code will group the elements in the form in a single box.

<!DOCTYPE **html**>

<**html**>

<**head**>

<**title**> Registration Form </**title**>

</**head**>

<**body**>

<**h2**>Registration Form</**h2**>

<**form** action="/register">

<**fieldset**>

<**label** for="firstname">First name :</**label**>

<**input** type="text" id="firstname" name="firstname">

<**br**><**br**>

<**label** for="lastname">Last name :</**label**>

<**input** type="text" id="lastname" name="lastname">

<**br**><**br**>

<**label** for="email">Email :</**label**>

<**input** type="email" id="email" name="email">

<**br**><**br**>

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

</**fieldset**>

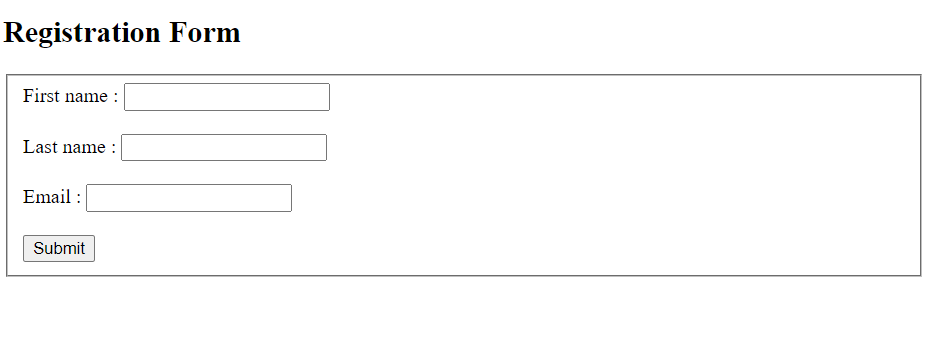
</**form**>

</**body**>

</**html**>

**Note: the fieldset is basically used in large form where you have to group the elements differently.**

When you have updated the code, save the file. The page should now look like this:



## Add a legend tag

What if there are multiple elements inside a form that needed to be grouped separately. How are we going to identify them?

By using the **legend** tag, we can add a title to the grouped element to recognize them.

The following code will group the elements and segregate them with the help of title.

<!DOCTYPE **html**>

<**html**>

<**head**>

<**title**> Registration Form </**title**>

</**head**>

<**body**>

<**h2**>Registration Form</**h2**>

<**form** action="/register">

<**fieldset**>

<**legend**>Personal Information</**legend**>

<**label** for="firstname">First name :</**label**>

<**input** type="text" id="firstname" name="firstname">

<**br**/><**br**/>

<**label** for="lastname">Last name :</**label**>

<**input** type="text" id="lastname" name="lastname">

<**br**/><**br**/>

<**label** for="email">Email :</**label**>

<**input** type="email" id="email" name="email">

</**fieldset**>

<**br**/><**br**/>

<**fieldset**>

<**legend**>Education Qualification</**legend**>

<**label** for="hdegree">Highest Degree :</**label**>

<**input** type="text" id="hdegree" name="hdegree">

<**br**/><**br**/>

<**label** for="Collegename">College name :</**label**>

<**input** type="text" id="Collegename" name="Collegename">

<**br**/><**br**/>

<**label** for="University">University :</**label**>

<**input** type="text" id="University" name="University">

<**br**/><**br**/>

</**fieldset**>

<**br**/><**br**/>

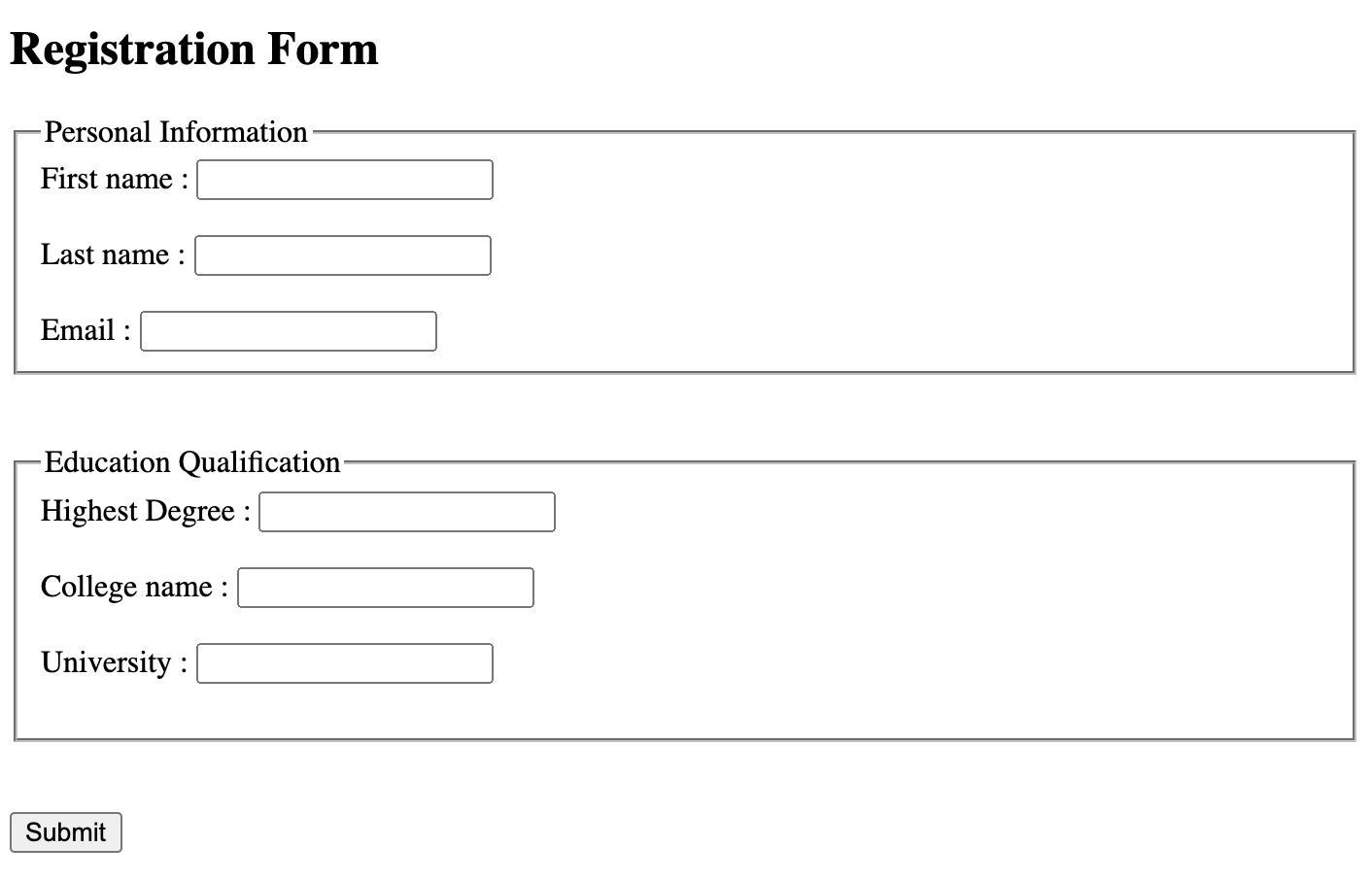
<**input** type="submit" value="Submit">

</**form**>

</**body**>

</**html**>

When you have updated the code, save the file. The page should now look like this:



## Summary

Congratulations! You have learned how to create an HTML form with input fields. Don't stop there -- we encourage you to explore other controls, such as radio buttons and check boxes.

## Tutorial details

**Author:** Ramesh Sannareddy

**Change log:**

| **Date** | **Version** | **Changed by** | **Change Description** |
| --- | --- | --- | --- |
| 2020-08-13 | 1.0 | Ramesh Sannareddy | Initial version created |
| 2021-09-01 | 1.1 | Sourabh Mohan | Update with new HTML Tags |
|  |  |  |  |

### © IBM Corporation 2020. All rights reserved.

<!DOCTYPE html>

<html>

<head>

<title>Registration Form</title>

</head>

<body>

<h2>Registration Form to use</h2>

<form action="/register">

<fieldset>

<legend>Personal Information</legend>

<label for="firstname">First Name :</label>

<input type="text" id="firstname" name="firstname">

<br><br>

<label for="lastname">Last Name :</label>

<input type="text" id="lastname" name="lastname">

<br><br>

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

<input type="email" id="email" name="email">

<br><br>

<br><br>

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

</fieldset>

<br><br>

<fieldset>

<legend>Education Qualification</legend>

<label for="hdegree">Highest Degree :</label>

<input type="text" id="hdegree" name="hdegree">

<br/><br/>

<label for="Collegename">College name :</label>

<input type="text" id="Collegename" name="Collegename">

<br/><br/>

<label for="University">University :</label>

<input type="text" id="University" name="University">

<br/><br/>

</fieldset>

<br/><br/>

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

</form>

</body>

</html>

# Reading: Module 3 Summary

In this module, you learned that:

* To structure a document, HTML5 uses elements like:
  + <div>
  + <header>
  + <footer>
  + <section>
  + <body>
  + <heading>
* HTML5 uses the <input> tag to allow users to input information. It has many different types, including:
  + Color
  + Date
  + Datetime-local
  + Email
  + Number
  + Range
  + Search
  + URL