ASP.Net Form Validation Using jQuery

* [](https://www.c-sharpcorner.com/members/vithal-wadje)

* [Vithal Wadje](https://www.c-sharpcorner.com/members/vithal-wadje)

* Feb 03, 2021

* 198.9k

* 11

* 8

[**FormValidationUsingJquery.zip**](javascript:void(0);)

[Download Free .NET Java APIs for Office Documents](https://www.e-iceblue.com/download/free-products.html/?aff_id=100)

**Background**

In my [previous article](https://www.c-sharpcorner.com/UploadFile/0c1bb2/Asp-Net-form-validation-using-javascript/) we had learned how to do validation Using JavaScript with a basic introduction. There are many disadvantages of using JavaScript that we will discuss later. In this article we will learn how to do ASP.Net form validation using jQuery .

If you are a beginner or student then please read my previous article:

* [ASP.Net Form Validation Using JavaScript](https://www.c-sharpcorner.com/UploadFile/0c1bb2/Asp-Net-form-validation-using-javascript/)

I hope you have read the preceding article. Let us now start from the creation of the function in jQuery.  
  
**Note**

* To use jQuery in an ASP.Net form you need to add the reference for the jQuery library that is nothing but a JavaScript file
* You can download the latest  jQuery library from <http://jquery.com/download/>

**Creating the function in jQuery.**  
  
The function is created in jQuery using the following syntax.  
  
**Syntax**

1. $(document).ready(function ()
2. {
3. $('#btnSave').click(function ()
4. {
5. })
6. });

In the preceding syntax, the function is the keyword provided by the jQuery to declare a function followed by a "$" (dollar) symbol and the "$('#btnSave').click" means that this function is called on the ASP.Net button click that has the id "btnSave".

**Example**

1. $(document).ready(**function** ()
2. {
3. $('#btnSave').click(**function** ()
4. {
5. alert("button is clicked")
6. })
7. });

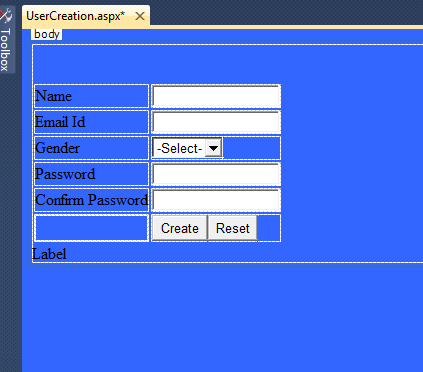
I hope you now understand the basics of validation in jQuery. Let us now create the one sample web application that demonstrates how to do validation.  
  
Let us first create the web application with two web pages as in the following:

1. "Start" - "All Programs" - "Microsoft Visual Studio 2010"
2. "File" - "New Website" - "C# - Empty website" (to avoid adding a master page)
3. Give the web site a name, such as Validation or whatever you wish and specify the location
4. Then right-click on the solution in the Solution Explorer then select "Add New Item" - "Default.aspx page" (add two pages).

We are adding two web pages because our requirement is, in the first web page there is form data to be filled in by the user and only after validating the form data, the form will be redirected to the next page.  
  
The first page source code <body> tag will look as in the following:

1. <body bgcolor="#3366ff">
2. <form id="form2" runat="server">
3. <br />
4. <br />
5. <div>
6. <table>
7. <tr>
8. <td>
9. Name
10. </td>
11. <td>
12. <asp:TextBox ID="txtUserId" runat="server"></asp:TextBox>
13. </td>
14. </tr>
15. <tr>
16. <td>
17. Email Id
18. </td>
19. <td>
20. <asp:TextBox ID="txtmail" runat="server"></asp:TextBox>
21. </td>
22. </tr>
23. <tr>
24. <td>
25. Gender
26. </td>
27. <td>
28. <asp:DropDownList ID="ddlType" runat="server">
29. <asp:ListItem Value="0">-Select-</asp:ListItem>
30. <asp:ListItem Value="1">Male</asp:ListItem>
31. <asp:ListItem Value="2">Female</asp:ListItem>
32. </asp:DropDownList>
33. </td>
34. </tr>
35. <tr>
36. <td>
37. word
38. </td>
39. <td>
40. <asp:TextBox ID="txt1" runat="server" TextMode="word"></asp:TextBox>
41. </td>
42. </tr>
43. <tr>
44. <td>
45. Confirm word
46. </td>
47. <td>
48. <asp:TextBox ID="txt2" runat="server" TextMode="word"></asp:TextBox>
49. </td>
50. </tr>
51. <tr>
52. <td>
53. </td>
54. <td>
55. <asp:Button ID="btnSave" runat="server" Text="Create"/>
56. <asp:Button ID="Button1" runat="server" Text="Reset" />
57. </td>
58. </tr>
59. </table>
60. <asp:Label ID="Label1" runat="server" Text="Label"></asp:Label>
61. </div>
62. </form>
63. </body>

Look at the preceding source code closely; see the ids of the controls that play the important role in JavaScript validation by reading the ASP.Net control values in JavaScript code.  
  
The design view of the preceding source code will look as in the following:



I hope you have created the same form for demonstration purposes as above.

**Reading ASP.Net Control Values in jQuery**

We can read ASP.Net control values in jQuery using the following syntax starting form the "$" symbol followed the "#" symbol and control Id as in the following:

1. $("#txtUserId").val(

How to add the .js file:

1. Right-click on Solution Explorer then select "Add New Item" then the in the "script.js" page rename the .js page as you wish, I have renamed it to "UserValidation.js".

Now declare the variable inside the function using the var keyword to read the ASP.Net control values by their ids and assign the values to the variable as:

1. $(document).ready(**function** ()
2. {
3. $('#btnSave').click(**function** ()
4. {
5. **var** Name, , gender, con, EmailId, emailExp;
6. Name = $("#txtUserId").val();
7. gender = $("#ddlType").val();
8. = $("#txt1").val();
9. con = $("#txt2").val();
10. EmailId = $("#txtmail").val();
11. emailExp = /^([a-zA-Z0-9\_\.\-])+\@(([a-zA-Z0-9\-])+\.)+([com\co\.\**in**])+$/;
12. })
13. });

In the code above I have taken the ASP.Net control's values in variables so it cannot be repeated again and again in our function the emailExp variable holds the pattern of the email id in the form of a regular expression.  
  
Now add the condition to ensure that all controls have a value, if the values are not entered in the form control then it will show a message. The condition will be as follows:

1. **if** (Name == '' && gender == 0 && == '' && con == '' && EmailId == '')
2. {
3. alert( "Enter All Fields");
4. **return** **false**;
5. }

In the preceding condition, to ensure that the form control's values are blank the message "Enter All Fields" is shown to the user and finally we are returning false; that is very important.  
  
**Importance of returning false**  
It's very important to use the return false statement after the condition block that returns false so if validation determines that the business requirements are not met then the form cannot be submitted. If you do not return false then the message will be displayed to the user that all fields are required but the form will be posted back and it gives you the second page directly. Therefore the return false statement works similar to the Required Field validator of ASP.Net.  
I hope you understand the concept.  
  
The condition that checks both text boxes for the word are as in the following:

1. **if** ( != con)
2. {
3. alert( "word not match");
4. **return** **false**;
5. }

In the preceding condition, we are checking that the two texboxes have words, in other words the word and confirm word.  
  
Condition to determine if the email address is valid:

1. **if** (EmailId != '')
2. {
3. **if** (!EmailId.match(emailExp))
4. {
5. alert( "Invalid Email Id");
6. **return** **false**;
7. }
8. }

In the preceding condition, first we ensure that the email id is not blank then we match the email id entered into the text box to the Regular Expression that is saved in the emailExp variable.  
  
The entire function will be as follows:

1. $(document).ready(**function**() {
2. $('#btnSave').click(**function**() {
3. **var** Name, , gender, con, EmailId, emailExp;
4. Name = $("#txtUserId").val();
5. gender = $("#ddlType").val(); = $("#txt1").val();
6. con = $("#txt2").val();
7. EmailId = $("#txtmail").val();
8. emailExp = /^([a-zA-Z0-9\_\.\-])+\@(([a-zA-Z0-9\-])+\.)+([com\co\.\**in**])+$/;
9. **if** (Name == '' && gender == 0 && == '' && con == '' && EmailId == '') {
10. alert("Enter All Fields");
11. **return** **false**;
12. }
13. **if** (Name == '') {
14. alert("Please Enter Login ID");
15. **return** **false**;
16. }
17. **if** (gender == 0) {
18. alert("Please Select gender");
19. **return** **false**;
20. }
21. **if** ( == '')
22. {
23. alert("Please Enter word");
24. **return** **false**;
25. }
26. **if** ( != '' && con == '')
27. {
28. alert("Please Confirm word");
29. **return** **false**;
30. }
31. **if** ( != con)
32. {
33. alert("word not match");
34. **return** **false**;
35. }
36. **if** (EmailId == '')
37. {
38. alert("Email Id Is Required");
39. **return** **false**;
40. }
41. **if** (EmailId != '')
42. {
43. **if** (!EmailId.match(emailExp)
44. {
45. alert("Invalid Email Id");
46. **return** **false**;
47. }
48. }
49. **return** **true**;
50. })
51. });

* Add the reference of the external jQuery file and the jQuery Library file into the Head section of the ASP.Net form

Just drag the .js file from the Solution Explorer to the Head section of the ASP.Net form that automatically adds the file reference path, it will look as in the following:

1. <head id="Head1" runat="server">
2. <script src="jquery-2.0.3.js" type="text/javascript"></script>
3. <script src="UserValidation.js" type="text/javascript">
4. </script>
5. </head >

* Calling jQuery function

To call the jQuery function on the ASP.Net button we do not need to call the function in the ASP.Net button's OnClientClick property that we have called in JavaScript because whenever we write the code inside the "$(document).ready(function()" function we just need to specify the control event in which the event of the control of the jQuery function will fire.

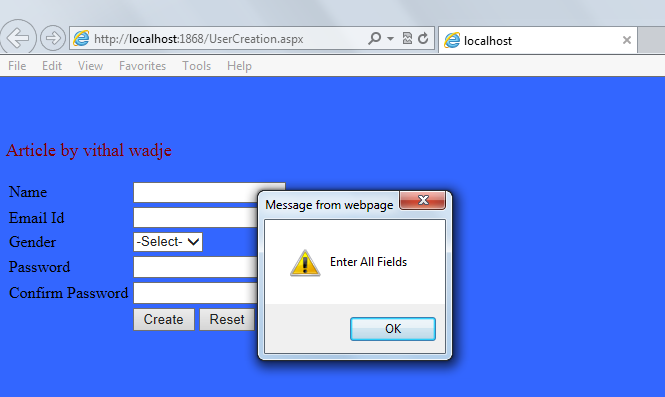
In the following example, I have called the function on the button click event that I have specified by using the control Id.

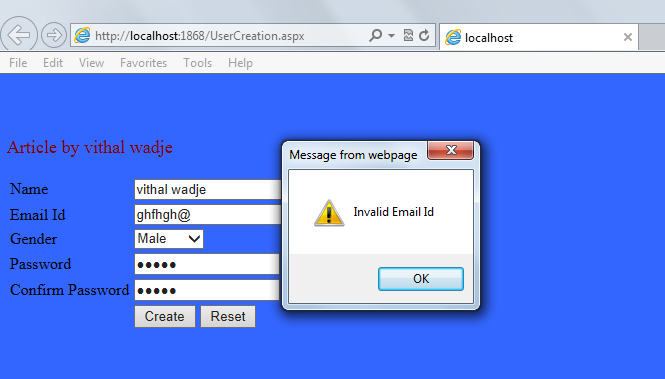
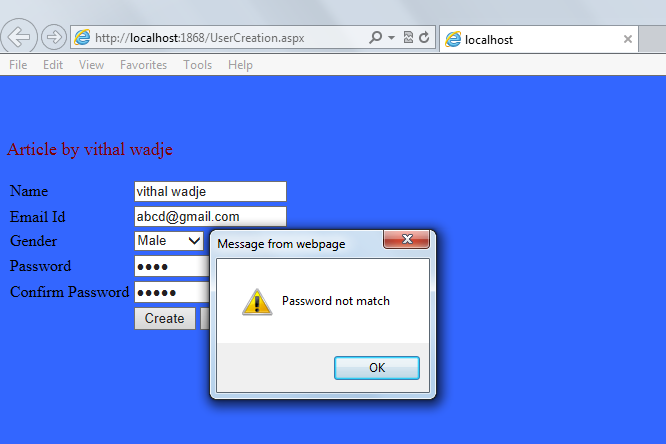
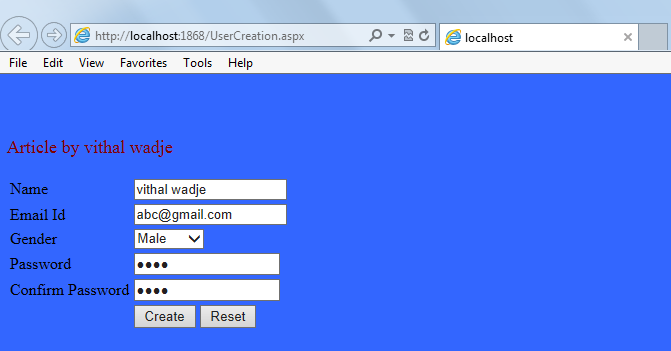
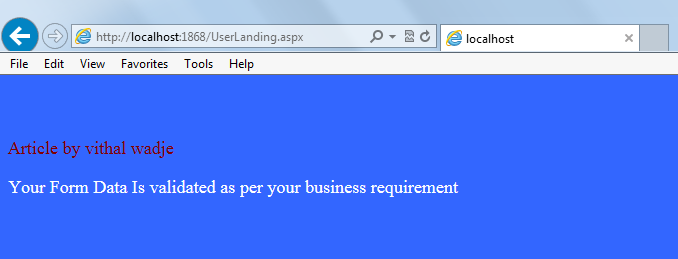
**Example**

1. $(document).ready(**function** ()
2. {
3. $('#btnSave').click(**function** ()
4. {
5. })
6. });

I hope you have understood it .  
  
In our demo application there are two ASP.Net pages, "UserCreation.aspx" and "UserLanding.aspx" along with the "UserValidation.js" JavaScript file and the Ajax libarary JavaScript file. In the "UserCreation.aspx" page the user enters the form details and then only after validating the details the page is redirected to the "UserLanding.aspx" page.  
  
Use the following code in the create button:

1. **protected** **void** btn\_Click(**object** sender, EventArgs e)
2. {
3. Response.Redirect("UserLanding.aspx");
4. }

In the code above, only after validating the form data, the page is redirected to the "UserLanding.aspx" page.  
  
Now run the ASP.Net web application and click on the "Create" button without inserting any data in the form. It will then show the following alert message.  
  
  
  
In the preceding screen you clearly see that even I have written the code on the create button to redirect to the next page but it will not be redirected because the form data is blank and it does not satisfy our validation condition that we set.

In other words, it's clear that the validation is done at the client-side in the browser level and only validates the data; it will execute the server-side code.  
  
Now enter the invalid Email Id, it will show the following message:  
  
  
  
Now enter the word that does not match the confirm word, it will show the following message:  
  
  
  
Now enter the valid details.  
  
  
  
Now click on the "Create" button; it will redirect to the next page as in the following:  
  
  
  
**Note**  
For detailed code please download the zip file attached above.  
  
**Summary**  
From all the examples above we see how to validate the form data using jQuery. I hope this article is useful for all students and beginners. If you have any suggestion related to this article then please contact me. My next article explains the advantages and disadvantages of validating the form data using jQuery and JavaScript.