JavaScript Forms

JavaScript Form Validation 검증

HTML form validation can be done by a JavaScript.

If a form field (fname) is empty, this function alerts a message, and returns false, to prevent the form from being submitted: 막다.

JavaScript Example

function validateForm() {  
    var x = document.forms["myForm"]["fname"].value;  
    if (x == null || x == "") {  
        alert("Name must be filled out");  
        return false;  
    }  
}

The function can be called when the form is submitted:

HTML Form Example

<form name="myForm" action="demo\_form.asp" **onsubmit="return validateForm()"** method="post">  
Name: <input type="text" name="fname">  
<input type="submit" value="Submit">  
</form>

[Try it Yourself »](http://www.w3schools.com/js/tryit.asp?filename=tryjs_validation_js)

상위처럼 사용자가 정의해서 폼검사를 할수 수 도 있지만 아래처럼 브라우저에서 해주 줄 수 있다.

HTML Form Validation

HTML form validation can be performed automatically by the browser:

수행하다.

If a form field (fname) is empty, the **required** attribute prevents this form from being submitted:

HTML Form Example

<form action="demo\_form.asp" method="post">  
  <input type="text" name="fname" **required**>  
  <input type="submit" value="Submit">  
</form>

[Try it Yourself »](http://www.w3schools.com/js/tryit.asp?filename=tryjs_validation_html)

|  |  |
| --- | --- |
| **Note** | HTML form validation does not work in Internet Explorer 9 or earlier. |

Data Validation 유효성검사

Data validation is the process of ensuring that computer input is clean, correct, and useful. 보장 프로세스(실행단위)

Typical validation tasks are: 일반적인 유효성 작업

* has the user filled in all required fields?
* has the user entered a valid date?
* has the user entered text in a numeric field?

Most often, the purpose of data validation is to ensure correct input to a computer application. 대부분의 경우

Validation can be defined by many different methods, and deployed in many different ways. 배포

**Server side validation** is performed by a web server, after input has been sent to the server. 수행된다.

**Client side validation** is performed by a web browser, before input is sent to a web server.

아래 예제를 확인해 보자.

<html>

<head>

<title>폼 유효성 검사 : 아이디 및 암호 체크</title>

<style type="text/css">

div, td, input { color:Navy; font-size:9pt; font-family:"맑은 고딕" Verdana 굴림; }

</style>

<script language="javascript" type="text/javascript">

function CheckForm()

{//A테그와 같이 Form테그는 미리 이벤트가 정의되어 있어서 제정의를 통해서 미리처리할 수 있다.

//[1] 아이디 텍스트박스에 접근해서 txtUserID 변수로 객체 담기

var txtUserID = document.FrmLogin.txtUserID;

if (txtUserID.value == "" ||

!(txtUserID.value.length >= 3 && txtUserID.value.length <= 12))

{ // 아이디가 입력되지 않았다면,

window.alert("아이디를 입력하시오.");

document.FrmLogin.txtUserID.focus(); // 해당 객체에 포커스

document.getElementById("txtUserID").select();//해당 객체의 텍스트 선택

return false; // 현재 submit 이벤트를 중지

}

else

{

document.FrmLogin.action = "./asp/Default.aspx";

document.FrmLogin.submit(); // 폼의 내용을 action 속성의 URL로 전송

}

}

</script>

</head>

<body>

<form id="FrmLogin" name="FrmLogin" action="asp/Default.aspx" method="post">

<div align="center">

<table border="1" width="300">

<tr><td align="right">아이디:</td>

<td><input type="text" id="txtUserID" name="txtUserID" /></td>

</tr>

<tr><td align="right">암호:</td>

<td><input type="password" id="txtPassword" name="txtPassword" /></td>

</tr>

<tr><td colspan="2" align="center">

<input type="button" value="로그인" onclick="CheckForm();" />

<input type="reset" value="취소" />

</td>

</tr>

</table>

</div>

</form>

</body>

</html>