Lecture 11: HTML Form Validation
ICT1153

Buddhika Gayashani Jayaneththi
Department of ICT
Faculty of Technology
University of Ruhuna

Encoding Form Data

- The **enctype** attribute specifies how the form-data should be encoded when submitting it into the server.

- enctype attribute can be used only if the **method="post"** is used.

Syntax

- <form enctype="*value*">

Default value

- **application/x-www-form-urlencoded**:

- All characters are encoded before sent (spaces are converted to "+" symbols, and special characters are converted to ASCII HEX values)

ASCII HEX Values - Example

ASCII Value	HEX value
!	21
"	22
#	23
$	24
%	25
&	26
'	27
(	28
)	29
*	2A

Encoding Form Data - Default

- Spaces: '+',

- Non-alphanumeric (special characters) :Ex- '%2A',

(a percentage sign and two hexadecimal digits representing the character)

- Line breaks, as in multi-line text field values: **' %0D%0A '**

- The fields are listed in the order they appear in the document with the name separated from the value by **'='** and the pairs separated from each other by `&'

- Fields with null values may be omitted

- In particular, unselected radio buttons and checkboxes should not appear in the encoded data, but hidden fields with VALUE attributes should present

Encoding Form Data…..

Example

Faculty	Technology Faculty
University	University of Ruhuna Matara

After encoding

Faculty=Technology+Faculty&University=University+of+Ruhuna%0D%0AMatara

Excercise

Write the encoded statement of the following form data

ID	001
Course	Internet & Web Development

Answer :

ID=001&Course=Internet+%26%0D%0AWeb+Development

HTML Form Validation

Form Validation

- **Validation :** The process of checking data against a standard or requirement.

Form Validation

- It forces users to fill out all required fields in a web form.
- Checks the integrity of the information provided.
  - Ex: The credit card has not expired

Validate Field Emptiness

Html tag

<input type="text" id="txtname"/>

1. Validate text field using JavaScript

```
function validateForm(){
    var name=document.getElementById("txtname").value;
    if(name== null || name== ""){
        alert("You must enter your name");
    return false;
    }
}
```

Validate Field Emptiness

1. Validate text fields using "required"

Html tag

<input type="text" id="txtname" required/>

StudentId |

Submit

:exclamation: Please fill out this field.

Validate Field Emptiness

```html
<html>
<body>
<form name="myForm" action=""
onsubmit="return validateForm()" method="post">
Name: <input type="text" name="fname" id="txtname">
StudentId <input type="text" name="sId" id="txtSID" required>
<input type="submit" value="Submit">
</form>
<script>
function validateForm(){
var name=document.getElementById("txtname").value;
if(name== null || name== ""){
alert("Name should be filledout");
return false;
}
}
</script>
</body>
</html>
```

Validate Field Emptiness

➢ The purpose of a form validation script is to return a boolean value (true or false) to the onsubmit event handler.

➢ A value of true means that form will be submitted while a false value will block the form from being submitting.

➢ The validateForm function tests the form input against our conditions, returning a value of **true** if the form is to be submitted (when all tests have been passed) or **false** to abort (cancel) the form submission.

Characters Only

JavaScript function

```
function allLetter()
{
 var letters = /^[A-Za-z]+$/;
 var name=document.getElementById("txtname").value;
 if(name.match(letters))
   {
    return true;
   }
 else
   {
   alert("You can enter characters only");
   return false;
   }
}
```

Characters Only Example

```html
<html>
<body>
<form name="myForm" action=""
onsubmit="return allLetter()" method="post">
Name: <input type="text" name="fname" id="txtname">
<input type="submit" value="Submit">
</form>
<script>
function allLetter()
  {
   var letters = /^[A-Za-z]+$/;
   var name=document.getElementById("txtname").value;
   if(name.match(letters))
     {
      return true;
     }
   else
     {
     alert("Enter characters only");
     return false;
     }
  }
</script>
</body>
</html>
```

Numbers Only

JavaScript function

```javascript
function allNums()
  {
    var numbers = /^[0-9]+$/;
    var mobile=document.getElementById("mobilenum").value;
    if(mobile.match(numbers))
    {
      return true;
    }
    else
    {
      alert("Enter numbers only");
      return false;
    }
  }
```

Numbers Only

```html
<html>
<body>
<form name="myForm" action=""
onsubmit="return allNums()" method="post">
Mobile No: <input type="text" name="mobile" id="mobilenum">
<input type="submit" value="Submit">
</form>
<script>
function allNums()
  {
   var numbers = /^[0-9]+$/;
   var mobile=document.getElementById("mobilenum").value;
   if(mobile.match(numbers))
     {
      return true;
     }
   else
     {
     alert("Enter numbers only");
     return false;
     }
  }
</script>
</body>
</html>
```

Numbers & Characters Only

<u>JavaScript function</u>

```javascript
function printNumChars()
  {
    var numchar =  /^[0-9a-zA-Z]+$/;
    var address=document.getElementById("txtaddress").value;
    if(address.match(numchar))
      {
        return true;
      }
    else
     {
       alert("Special characters are not allowed");
       return false;
     }
  }
```

Numbers & Characters Only

```html
<html>
<body>
<form name="myForm" action=""  method="post">
Address: <input type="text" name="address" id="txtadd">
<input type="button" value="Submit" onclick="numschars()">
</form>
<script>
function numschars()
  {
   var numchar =  /^[0-9a-zA-Z]+$/;
   var address=document.getElementById("txtadd").value;
      if(address.match(numchar))
     {
         return true;
     }
   else
     {
     alert("Special characters are not allowed");
     return false;
     }
  }
</script>
</body>
</html>
```

Questions