INT222 - Internet Fundamentals

Week 10: Date Object, Event and JS Form Validation

(Ver 1.1)

Agenda

- JS built-in Object - Date
- DOM Event and Event Handler
- Form Validation

Date Object

- Enables basic storage and retrieval of dates and times.
- Creates a Date object with current date and time:

```
var myDate = new Date();
```

- date string:

```
alert("The date is  " + myDate);
```

Will show the date string:

The date is Mon Mar 10 2014 09:02:37 GMT-0400 (Eastern Standard Time)

The get… Methods of Date Object

- getMonth() method
  - Returns number of **0** through **11**
    - Represent month of **January** through **December** correspondingly
  - e.g.

```
var myMonth = (new Date()).getMonth();
alert(myMonth); // The myMonth is  6  which is,  July
```

- getDate() method
  - returns number of 1 31
  - e.g.

```
var myDay = (new Date()).getDate();
alert(myDay ); //
```

The get… Methods of Date Object

- getDay() method
  - returns number of **0** for **Sunday**, **1** for **Monday**, …
  - e.g.

```
var myDayOfWeek = (new Date()).getDay();
alert(myDayOfWeek  );
```

- getFullYear() method
  - returns a 4 digit year
  - e.g.

```
var myYear = (new Date()). getFullYear();
alert(myYear );  // 2014
```

The get… Methods of Date Object

- getHours() method
  - returns a number of 0 to 23
- getMinutes() method
  - returns a number of 0 to 59
- getSeconds() method
  - returns a number of 0 to 59
- e.g.

```
var myDate = new Date();
var myHour = myDate.getHours();
var myMinutes = myDate.getMinutes();
var mySeconds = myDate.getSeconds();
alert(myHour + ":" +  myMinutes  + ":" + mySeconds); // 10:9:35
```

The get… Methods of Date Object

- getMilliseconds() method
  - Gets the milliseconds of a Date, 0 to 999. .
  - e.g.

```
var myMillSec = (new Date()).getMilliseconds();
alert(myMillSec );
```

Resource & Reference

- Unix (epoch) time:
  - http://www.w3schools.com/jsref/jsref_gettime.asp
- JS Date Constructor
  - https://developer.mozilla.org/en-US/docs/Web/JavaScript/Reference/Global_Objects/Date
- JS Date set… Methods

Events

- An event occurs when a user clicks on a link or a button in a form
- In general everything that happens in a browser may be called an event.
- Every element on a web page has certain events which can trigger a JavaScript function.
- JavaScript needs a way of detecting user actions so that it knows when to react. It also needs to know which functions to execute.

Common Events

- Events triggered by user actions.
  - e.g.
    - When a user clicks the mouse
    - When a user strokes a key
    - When the mouse moves over an element
    - When an input field is changed
    - When an HTML form is submitted
- Events that are not directly caused by the user.
  - e.g.
    - When a web page has finished loading
    - When an image has been loaded
- Events category
  - Mouse events, keyboard events, HTML frame/object events, HTML form events, user interface events, touch events, …

Event Handlers

- Event Handlers are used to manipulate documents.
- An event handler is used in order to execute a script when an event occurs.
- The event handler has a prefix "on" followed by the event name.
- For example, the event handler for the click event is onclick.

Event Handlers

- Event handlers are divided into:
  - Interactive event handler
    An interactive event handler depends on the user doing something to a document such as moving a cursor over an object.
  - Non-interactive event handler
    A non-interactive event hander execute automatically depending on events such as onload.

Event Handler Examples

- Using event handlers to change the elements of a document by
  - writing a line of code in the value of the event handler

```
<input type="button" name="MyButton" value="New Button!"
        onclick="window.open('mywindow.html', 'MyWin')" />
```

  - writing the script as a function in the head section of the document and then calling the function from the event handler (refer to the given examples).

```
<input type='checkbox' name='system_type' value='4'
        onclick='commonSense()'/>Unix
```

Creating Event Handler

- To create an event handler for an HTML tag:
  - add the event handler attribute to the tag.
  - write the JavaScript code in quotation marks as the attribute value..

Creating Event Handler

- The general syntax is

```
<htmltag eventHandler="JavaScript Code">
```

  - where htmltag is an HTML tag/element
  - eventHandler is the name of the event handler
  - JavaScript Code is a set of JavaScript statements.
  - The JavaScript statement(s) are executed when the event occurs - such as the user clicks on a link or a button.

Creating Event Handler

- Notes:
  - The event handlers in HTML must be enclosed in quotation marks
  - Alternate double quotation marks with single quotation marks:
- e.g.

```
<input type="button" name="MyButton" value="New Button!"
        onclick="alert('some text')" />
```

- You can also use \' or \" to achieve the same result.

```
<input type="button" name="MyButton" value="New Button!"
        onclick="alert(\"some text\")" />
```

Using the HTML DOM

- To create a event handler to an element, you can assign the event handler to the element as an attribute in HTML

```
<button onclick="displayDate()">Try it</button>
```

- The HTML DOM allows you to assign the event handler to the element using JavaScript

```
<script>
  document.getElementById("myBtn").onclick=function(){displayDate()};
  // document.getElementById("myBtn").onclick=displayDate; //the same

  function displayDate() {
    document.getElementById("demo").innerHTML = Date();
  }
</script>
```

Event Handler Examples

- onchange:

  occurs when the content of a field changes.

  - Applies to :

    select, text, input elements

  - Example:

    js_onchange.html

Event Handler Examples

- onclick
  - Occurs when the user has pressed and released a mouse button (or keyboard equivalent) on an element.
  - Applies to:
    button, document, checkbox, link, radio, reset, submit
  - Example:
    js_onclick.html

Event Handler Examples

- ondblclick
  - Occurs when the user has double-clicked a mouse button on an element.
  - Applies to the following HTML tags:
    document, image button elements, link
  - Example:
    js-ondblclick.html

Event Handler Examples

- onfocus
  - Occurs when the user has given focus to an element.
  - Applies to

    button, checkbox, file, password, radio, reset, select, submit, text, textarea, window.
  - Example:

    js-onfocus.html

Event Handler Examples

- onload
  - Occurs when a document or other external element has completed downloading all data into the browser.
  - Applies to
    image, window.
  - Example:
    js-onload.html

Event Handler Examples

- onmouseout
  - Occurs when the user has rolled the mouse out of an element.
  - Applies to

    image, window.
  - Example:

    js-onmouseout.html

Event Handler Examples

- onmouseover
  - Occurs when the user has rolled the mouse on top of an element.
  - Applies to

    image, window.
  - Example:

    js-onmouseover.html

Event Handler Examples

- onresize
  - Occurs when the user has resized a window or object.
  - Applies to
    window.
  - Example:
    js-onresize.html

Validation General Guidelines for Raw Data

The following is a list of guidelines for validating raw data. Some or all of these guidelines may apply when validating - it all depends on the application.

- Presence or Absence Test
  - The presence or absence test is used to determine if a field has been completed.
- Test to determine if a field contains
  - a numeric value
  - an alphabetic value
  - an alphabetic - Upper case
  - an alphabetic - Lower case

Validation General Guidelines for Raw Data

- Test to determine if a field is
  - Positive
  - Negative
  - Zero
- Value Test
  - The value test is used to determine if a field has a specific value or code.
- Range Test
  - The range test is used to determine if a value entered is within a specific range (inclusive or exclusive)

Validation General Guidelines for Raw Data

- Reasonableness Test
  - The reasonableness test is used to determine if a value entered is reasonable based on other information supplied or information available to us. This test needs to be review periodically.
- Consistency Test MULTIPLE FIELD(s)
  - The consistency Test is used to determine if a value entered is consistent with other information entered.
- Check Digit Test – See how it works
  - The check digit test is used to determine if for example, a credit card number or a Driver license number is valid.

Form Validation Using JavaScript

JF Form Validation: input type="text"

- Simple Form validation for input type="text"
  - using document.formName.elementName
    - js-validation-example-Byname/
  - using document.getElementById("elementId") method
    - js-validation-example-elementById/
  - using document.getElementsByName("elementName") method
    - js-validation-example-elementByName/
  - using document.forms[index].elements[index] - Not recommended
    - js-validation-example-formelementIndex/
  - using the this key work
    - js-validation-example-with-this/
- Learning JS validation coding from these examples

JF Form Validation: input type=" checkbox"

- Example
  - js-validation-checkbox/
- Checkbox logic
  - Get the number of the checkboxes using length
  - Loop to check which one was checked
  - To determine which one is checked - if any

```
document.example.system_type[i].checked == true;
  //  checked
document.example.system_type[i].checked == false;
  //  not checked
```

JF Form Validation: input type="radio"

- Example
  - js-validation-radio/
- radio logic
  - Get the number of the radio using length
  - Loop to check which one was checked
  - To determine which one is checked - if any

```
document.example.system_type[i].checked == true;
  //  checked
document.example.system_type[i].checked == false;
  //  not checked
```

JF Form Validation: select Fields

- Select with single option. Example:
  - js-validation-select-single/
- Select options logic
  - Get the selectedIndex:

    var x = document.example.whatToDo.selectedIndex;
  - If selectedIndex == -1
    - None are selected
  - If the selectedIndex is NOT -1
    - document.example.whatToDo.options[x].value;
      - for the value
    - document.example.whatToDo.options[x].text;
      - for the text

Text vs Value

- For the <select> <option> elements – dropdown list

```
<select>
  <option value="This is a value">This is the text</option>
  <option value="This is another value"  selected>
        This is another text
  </option>
</select>
```

- Any other form fields/controls have text attribute?

JF Form Validation: select Fields

- Select with multiple options. Example:
  - js-validation-select-multiple/
- Select options logic : **multiple="multiple" **
  - Get the number of the select options using length

    document.example.whatToDo.options.length;
  - Loop to check which one was selected
  - To determine which one is selected

    if (document.example.whatToDo[i].selected == true) // selected

    document.example.whatToDo[i].value

    document.example.whatToDo[i].text

JF Form Validation: textarea Fields

- Form validation for textarea elements
  - js-validation-textarea/

```
function checkForm() {
        var textareaLength = document.example.comments.value.length;
        var messages="<p>No. of characters in textarea = <mark>" +
                              textareaLength + "</mark></p>";
        if (textareaLength==0) {
            messages+= "<p>You did not type any text in the textarea</p>";
            showErrors(messages);
            return false; // return false - allow for changes - form not submitted
        }
        else {
            messages +="<p>You typed</p><p>" + document.example.comments.value +
"</p>";
            showErrors(messages);
            return false;  // return true - form will be submitted
        }
    } // End of function
```

Resourceful Links

- Event - Web API Interfaces | MDN
- Onclick vs addEventListener

Thank You!