27/3/20

DHTML - Dynamic HTML

* It is a powerful language. comes with predefined objects using which we could able to perform any type of CRUD operation on any DOM element on its corresponding CSS property.

* using js, we could able to generate the complete HTML page with all dynamic data as and when required.

* It provides pre-defined methods and objects using which we could able to handle any type of events being generated on the page.

* It provides pre-defined methods and objects using which we could able to read, write or form on the HTML input elements.

Note:

'document' and 'window' are two pre-defined object by default available in every web page is actually holds the complete DOM structure on which we can perform any type of CRUD operation

Event Handling

* Event: Any action been performed on a page is called an event.

* The process of performing a particular job bases on a particular action been generated on a page is called event handling.

* following are the types of action might get generated on a page.

① click
② dbclick
③ mouseover
④ drag
⑤ drop
⑥ Keypress
⑦ keydown
⑧ focus
⑨ blur etc.

* following are the two ways we could able to implement Event Handling:-

→ Static Event Handling.
→ Dynamic Event Handling.

① Static Event Handling.

In this way, using HTML pre-defined static event attributes, we statistically specify the particular is function need to be invoked when a particular action happens on a page.

following are the pre-defined HTML attributes

(i) onclick
(ii) onmouseover
(iii) onfocus
(iv) onchange
(V) onblur
(vi) on keyup
(vii) onkeydown
(viii) onmouseout
(xi) ondbclick etc.

Ex:

<div onclick = "displayData()"> - - - - </div>

// when there is a click on the above div element, method "displayData()" get automatically.

② Dynamic Event Handling:

* java script provides a feature through which we can dynamically specify what method need to be invoked when a particular action happens on a DOM element.

"AddEventListener" It is a pre-defined method been using which we can handle action pre-defined on a page dynamically.

Syntax:

Element.addEventListner("type of event",
                        <call-back method>)

* This method attaches an event hander to an element without corresponding existing event handling.

* you can add many event handlers to any element

Window and Document:

* The two pre-defined object been supported in JS using through which we could perform any type of operation on any Html element of its corresponding css properties

* following are the different ways are

could refer a dom element on a page.

* document.getElementById(): using which we could refer to a Dom element on page using its unique ID.

* document.getElementByclassName(): using which we could refer to a dom element on page using its name.

* document.getElementsByTagName(): using which we could refer to dom element of page using its tag name.

* document.querySelector(): A method by supported in HTML5 using which we can refer to any dom element dynamically using js. maybe through by Id, by classname or tag name.
  document.querySelector(#idname)/(.class name)/(tag name)

following are the methods can be applied on element references:

* element.setAttribute("attribute Name", "Value") - using which we could able to set any attribute dynamically to a Dom element.

* element.getAttribute("Attribute Name") - using which we can get corresponding attribute value of any dom element.

* element.Style - using which dynamically any css property of any Dom element can be retrieved or updated.

* element.appendchild (<element>) - using which we could able to append an element to an Existing dom element page.

* document.create Element ("<tag Name>") - using which we could able to create any dom element using javascript.

Reading Values from Input elements:

* Js provides pre-defined methods and objects through which we could able to read data and value from input elements

→ Element.Value // returns value from input elements and text area.

Elements.option // Returns array of options as object from a drop down element.

→ Element.SelectedIndex - // returns current selected index of dropdown element.

→ Element.checked - Returns true / false of non-selected state of current selected or checkbox etc.