158.258 - Laboratory Practicals

- Session 7 -

Getting acquainted with Javascript
2018

Create a basic web page

1. Create a basic web page that includes some elements.

2. Add a **<script>** block to include some additional content using : **document.write()**

Calling Javascript functions

Create a page that contains **two buttons**.

Arrange the button to call a javascript **function `popup()`** that you've written.

The function `popup(x)` is really simple - it just displays the parameter *x* in an alert box.

Write this in two ways:

1. With the function inside a <script> … </script> block

2. Call function from an external file.

Getting content from a text box

Create a page with a **text box** and a **submit button**

- the text box has an id='reply' tag
- when the submit button is pressed, display the content of the text box in an 'alert()' popup.

To get the value from the text box, you'll need to use the *getElementById(id-Tag)* function.

Once you've got access to the right element, you can extract content:

e.g.

```
        v = document.getElementById('reply').value
```

Or update the contents of an existing element:

```
document.getElementById('idTag').innerHTML = 'new value'
```

where 'idTag' is the id of some element (e.g. a <p>)

Conditions

- **if statements**

  Modify this example so it:

  1. Only displays a pop-up if the textbox value is contains 'fred'
  2. It would have a counter that counts the number of times the popup occurs and displays that along with the textbox value
  3. It uses a conditional expression rather than an *if-statement* to convert *fred* to *FRED*

- **for loops**

  1. Add a paragraph heading with the id of 'output' and display all the numbers from 1 to 10 using a for loop.
  2. Create an **text box** for the upper limit and display values from 1 to this limit
     - Using the function **parseInt(s)** to convert the string from the textbox.

  Checking the Numeric Conversion:

  - Only run the **for loop if** the result from *parseInt()* is a number.

  If parseInt() returns **NaN**, display an alert box with "Not a valid number: " and then the textbox contents.

Arrays

Make a page that has a **textbox** and a **two buttons**:

- ***Add*** - Adds the textbox contents to the array.
- ***Show*** - Outputs all of the stored items in the array into a <h3> tag.

Then add these buttons:

- ***Clear*** - Deletes all the elements from the array.

- ***Delete-Last*** - Deletes the most recently added element.

Use Change-document-font to do other things

```
<h4>Change Body Font</h4>

<input type=submit value='Body Font is Arial  ' onClick="setFont('Arial'  )"> <br>
<input type=submit value='Body Font is Georgia' onClick="setFont('Georgia')"> <br>
<input type=submit value='Body Font is Times  ' onClick="setFont('Times'  )"> <br>
<input type=submit value='Revert Body Font    ' onClick="revertFont()"      > <br>
<input type=submit value='Show Changes        ' onClick="showChanges()"     >

<p><b>Changes</b></p>
<p id = 'changeList'> None yet </p>

<script>
originalFont=null  // Global - place to save original
fontChanges = Array();

function setFont(whichFont){
   if (originalFont == null) {
      originalFont = document.body.style.fontFamily;
   }
   document.body.style.fontFamily = whichFont
   fontChanges.push(whichFont)
}
function showChanges(){
   if (fontChanges.length == 0)
     alert("No changes")
   else {
      s = ''
      for (i=0; i<fontChanges.length; i++) {
        s = s + fontChanges[i] + '<br>'
      document.getElementById('changeList').innerHTML = s
      }
   }
}
```

Variable

- **Increment & display a global variable**

  1. **Putting a your buttons inside forms can cause unexpected side effects with Javascript.**

     The following four buttons should just increment the global 'count' and display it. The fourth button (a submit) doesn't act as expected.

     Try clicking the first three buttons in order and then the fourth one twice to identify the problem.

  2. **Both of these increment count (as expected)**

     1. type=Button outside Form

     2. type=Submit outside Form

3. Careful: using a submit button IN A FORM reloads page and so resets global variable

3. type=Button within Form

4. type=Submit within Form < Reloads page & resets count

4. Summary

If you're going to use buttons to just to trigger Javascript events, don't place them inside a form construct.

Only use a form when you're wanting to actually submit a set of fields to an external server.

. Edited 2018: Nazi Tabatabaei-Yazdi