Web Technologies – Exercises: Week 7

This week's exercise uses JavaScript DOM1 properties, methods and events in a scenario akin to the Minesweeper exercise (to give us an excuse to work together solving some of the issues) – it's in advance of Lecture 7 where the rest of the DOM1 material will be covered so it includes lots of JavaScript explanation ☺

Task 1:   Do the quiz.

... although the last question is from lecture 7/8 so the deadline is 2 weeks hence (as it's in-class test 1 next week too!)

Task 2:   Display information from click events and respond appropriately:

Develop your understanding of how to use events and event handlers within a table-based grid – you should be able to test *something* after each numbered step.

1) Create a valid XHTML page in a file called week7table.htm entitled "Week 7 clicks-in-a-table".

2) Write an HTML table with

   a) at least 3 columns and 3 rows

   b) containing example mines *e.g.* using "*" for mines

   c) adjacent mine counts as numbers (text)

   d) IDs on each cell like "**r1c1**" *etc.*

3) Add some CSS style *e.g.* for borders and text style.

4) Add an onclick event handler to the table which passes the `event` object to the function, *e.g.* **<table onclick="**`clickHandler(event)`**">** -- if you preview this and click on a cell the browser should give an error message until you do the next step...

5) Create a **<script>** tag containing the function definition, *e.g.*

   a) **<script type="text/javascript">**

   `function clickHandler(e) {`

   `}`

   </script>

6) Use alert to pop-up a message which says what the ID and contents of the cell are

   a) one `alert`, use string concatenation

   b) if the argument e is the event object then `e.target.id` is the cell's ID

   c) and `e.target.innerHTML` is the cell's contents

7) Add to the CSS **td {color:white;background-color:white;}** so the cell's text disappears.

8) Modify the `clickHandler` function so that it uses the CSS style object "color" property to reset the text colour to black *e.g.* `e.target.style.color = 'black';`

9) Remove the "`alert`" from 6 and add

    a) an if-test that checks to see if the cell's contents are "*" – a mine!

    b) if-so, `alert('Boom!');`

10) If the cell is "empty" (has no adjacent mines) can you write code that would change the colour in its neighbouring cells? A manual approach (lots of if-tests) would be fine here... you can think about automating it later for the game :-)

If you get this far you've created something which does parts of the

Don't forget to do some reading!

Dr James Denholm-Price