
Web Technologies – Exercises: Week 2

For this exercise we'll use some more advanced CSS and HTML features – as always the lecture notes are available on StudySpace for reference!

Outcomes:

By the end of the second week, you should have:

1. Completed the second weekly quiz.
2. Styled a data table using CSS, columns and selectors.
3. Built a CSS-based flag (rather than the <table>-based one from week 1).

Task 1: Do the week 2 quiz.

Look in the “Weekly Exercises” folder on StudySpace.

- **Note 1:** You must be enrolled on the StudySpace module (“Guest Access” is *not* OK.)
If you're not enrolled by next week's 5PM deadline, email me.
- **Note 2:** The quiz uses the University's “Perception” software, not StudySpace.

Task 2: Style a table of data using CSS.

You should have read some chapters in the “CSS” text by now ☺ In this exercise you should use CSS to add style to [this data table](#).

NB: Apologies if you are a keen footie fan I *know* the screenshots are out of date!

1. Take a copy of the [raw table page](#)¹.
 - Careful with “Save As”, the doc should be valid XHTML with lots of rows, columns & data.
2. Move the <style> block contents to an external style sheet, *e.g.* “week2league.css” and <link> it back in the HTML.
3. Add a 1 pixel width border to all of the cells in the table *except* the empty <td> elements in the top class=“header” row.

#	Team	Home			Away			GD	Pts						
		P	W	D	L	F	A								
1	Chelsea	6	3	0	0	7	2	2	1	0	4	1	8	16	
2	Everton	6	2	1	0	6	3	2	0	1	6	3	6	13	
3	Man Utd	6	2	0	1	9	5	2	0	1	5	4	5	12	
4	Man City	6	2	1	0	7	4	1	2	0	5	4	4	12	
5	Tottenham	6	1	2	0	4	3	2	0	1	7	5	3	11	
6	West Brom	6	3	0	0	6	0	0	2	1	2	5	3	11	
7	Arsenal	6	1	1	1	7	3	1	2	0	3	1	6	9	
8	Fulham	6	2	0	1	9	2	1	0	2	4	7	4	9	
9	Newcastle	6	2	1	0	4	2	0	2	1	4	6	0	9	
10	West Ham	5	2	1	0	5	1	0	1	1	0	3	1	8	
11	Swansea	6	1	1	1	5	5	1	0	2	5	4	1	7	
12	Stoke	6	1	2	0	3	1	0	2	1	3	4	1	7	
13	Sunderland	5	1	1	0	2	1	0	3	0	3	3	1	7	
14	Liverpool	6	0	1	2	3	6	1	1	1	6	6	-3	5	
15	Aston Villa	6	1	1	1	4	4	0	1	2	2	6	-4	5	
16	Wigan	6	0	1	2	3	6	1	0	2	2	5	-6	4	
17	Southampton	6	1	0	2	6	6	0	0	3	4	1	2	-8	3
18	Norwich	6	0	2	1	3	6	0	1	2	1	7	-9	3	
19	Reading	5	0	2	1	4	6	0	2	2	5	-5	-2		
20	QPR	5	0	1	1	0	5	0	1	2	3	6	-8	2	

#	Team	Home			Away			GD	Pts					
		P	W	D	L	F	A							
1	Chelsea	6	3	0	0	7	2	2	1	0	4	1	8	16
2	Everton	6	2	1	0	6	3	2	0	1	6	3	6	13
3	Man Utd	6	2	0	1	9	5	2	0	1	5	4	5	12
4	Man City	6	2	1	0	7	4	1	2	0	5	4	4	12

¹ <http://staffnet.kingston.ac.uk/~ku13043/WebTech/ex/week2league.html>

Web Technologies – Exercises: Week 2

- **Hint:** Make a general rule for the `<table>` and cells, `<td>` and `<th>`, then override it with a specific, contextual selector for the `<td>` elements in the header row.

4. The [original BBC site](#) where I “borrowed” the information used CSS classes to style the columns – *lots* of classes! There is a simpler way (but HTML only permits CSS borders, background, width and visibility properties to be set): Research online the function and syntax for `<colgroup>` and `<col>`.
5. Edit the HTML to add a `<colgroup>` element containing `<col>` elements for the Home and Away headed column groups (and any others necessary), identified using IDs.

6. Use CSS to give the Home and Away column groups different background colours using the `<col>` ID selectors (choose your own colour).

#	Team	Home					Away					GD	Pts	
		P	W	D	L	F	A	W	D	L	F			A
1	Chelsea	6	3	0	0	7	2	2	1	0	4	1	8	16

7. To improve readability of large data tables it’s useful to “zebra stripe” alternate rows with a grey background colour to enhance readability (called “zebra tables”) – often it’s done with *two* CSS classes or CSS3 selectors but it’s easy to do with just one class. The CSS class `odd` is in the example document. Add a CSS rule to zebra-stripe the table.

#	Team	Home					Away					GD	Pts	
		P	W	D	L	F	A	W	D	L	F			A
1	Chelsea	6	3	0	0	7	2	2	1	0	4	1	8	16
2	Everton	6	2	1	0	6	3	2	0	1	6	3	6	13
3	Man Utd	6	2	0	1	9	5	2	0	1	5	4	5	12

8. On the BBC site a class “divider” is used to add dotted lines to separate various groups of teams. Add a `class` or `ID` to the relevant rows in *your* HTML and CSS to reproduce that effect around Tottenham and above Norwich, something like the picture on the right:

#	Team	Home					Away					GD	Pts	
		P	W	D	L	F	A	W	D	L	F			A
1	Chelsea	6	3	0	0	7	2	2	1	0	4	1	8	16
2	Everton	6	2	1	0	6	3	2	0	1	6	3	6	13
3	Man Utd	6	2	0	1	9	5	2	0	1	5	4	5	12
4	Man City	6	2	1	0	7	4	1	2	0	5	4	4	12
5	Tottenham	6	1	2	0	4	3	2	0	1	7	5	3	11
6	West Brom	6	3	0	0	6	0	0	2	1	2	5	3	11
7	Arsenal	6	1	1	1	7	3	1	2	0	3	1	6	9
8	Fulham	6	2	0	1	9	2	1	0	2	4	7	4	9
9	Newcastle	6	2	1	0	4	2	0	2	1	4	6	0	9
10	West Ham	5	2	1	0	5	1	0	1	1	0	3	1	8
11	Swansea	6	1	1	1	5	5	1	0	2	5	4	1	7
12	Stoke	6	1	2	0	3	1	0	2	1	3	4	1	7
13	Sunderland	5	1	1	0	2	1	0	3	0	3	3	1	7
14	Liverpool	6	0	1	2	3	6	1	1	1	6	6	-3	5
15	Aston Villa	6	1	1	1	4	4	0	1	2	2	6	-4	5
16	Wigan	6	0	1	2	3	6	1	0	2	2	5	-6	4
17	Southampton	6	1	0	2	6	6	0	0	3	4	12	-8	3
18	Norwich	6	0	2	1	3	6	0	1	2	1	7	-9	3
19	Reading	5	0	2	1	4	6	0	0	2	2	5	-5	2
20	QPR	5	0	1	1	0	5	0	1	2	3	6	-8	2

9. Make sure it’s still valid XHTML & CSS.

If you have time to spare:

10. Look for a CSS3 alternative to using `class="odd"` to style the odd/even rows – is it compatible with every browser? If it is at least compatible with IE8, use it in your example to replace the `class="col"` stuff.

Web Technologies – Exercises: Week 2

(**Hint:** it's possible to remove all reference to `class="odd"` in the XHTML and use one or several rules in the CSS to achieve the same effect.)

11. Try to change other style properties on the `<col>` elements using CSS, such as the font-weight and border-style. Test in Firefox, Opera *and* Internet Explorer. Write about what you find beneath the table in your HTML document, *e.g.* produce a brief browser support table. (For example, which browser lets you change the border-style, and does it work properly?)

The [HTML 4 specification](#) defines [col and colgroup](#) and the [CSS2 specification](#) says [what styles can be changed](#), but not all browsers bother to follow the spec...

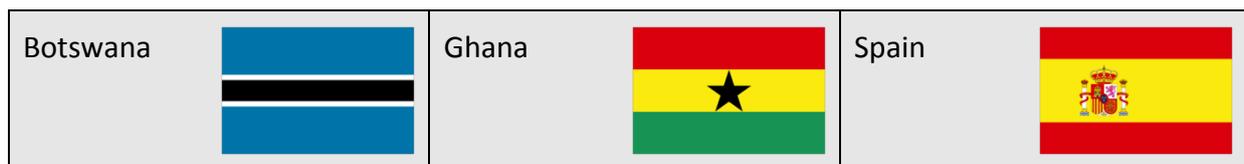
Note:

- Get used to using a development tool *other than* Notepad: TopStyle and Dreamweaver provide useful mechanisms for creating CSS; Notepad++ and HTML Kit give syntax-highlighting hints and are free to download at home; Dreamweaver and HTML Kit give you quick access to a preview window (but be wary of *which* rendering engine these tools use ... always test in a real browser!)

Task 3: Draw another flag using CSS.

You should have read some chapters in the “CSS” text by now ☺ In this exercise you should use CSS to lay-out the flag (*not* a `<table>`).

1. Pick a new flag from *e.g.* [Flags of the World](#), not the same as week 1's and one that includes **at least 4 elements** or **regions**², like the following examples:



NB: if you pick a flag with vertically-oriented elements you will need to use something like the CSS *positioning* properties that were in lecture 2 slides ... horizontally-oriented flags like the above can be done without CSS *position* (perhaps with creative use of borders and padding ☺).

2. Create an HTML page called `flag2.html` that uses several `<div>` elements, possibly one `` element and lots of CSS positioning to create the flag.

² Such as the three coloured regions and an image in Ghana or Spain, or the three regions and white borders from Botswana.

Web Technologies – Exercises: Week 2

- If your flag includes a complex image (like Spain's) try extracting just the image using a graphics tool like Paint, [Irfanview](#), Photoshop, The GIMP *etc.*
- b) You may *develop* the CSS using inline style but it should eventually reside in an external CSS file called [flag2.css](#)
- c) If the flag is “vertically oriented” then you could use **absolute** and/or **relative** positioning to position the flag's elements or **float** if you fancy doing some research.
- NB:** Whatever the layout of your flag this week, **don't use a table...**
- d) It's easiest to wrap the whole flag in a container element, like a `<div>` and put `<div>`'s for the elements of the flag inside the container.
- e) Centre the flag horizontally.
- **Hint:** Wrap it in a relatively-positioned container and centre that...
3. Like last week's exercise, include a page title that says something like
“The flag of Botswana by James (k1234567)”
and a level 1 heading element that includes the name of the flag and that links to the [Wikipedia](#) page for that country.
4. At the bottom of the HTML page include one `<address>` block with your name, KU number and course title on separate lines. Right-justify the text using CSS (but make sure that all of the CSS is in the style sheet.)
5. Make sure it's valid XHTML & CSS.

Note:

- You *could* use something like Dreamweaver to cheat with the design but then all you'd have learnt was “how to do it with Dreamweaver” *not* how to write the code ☹

This concludes the second week's exercises ... before next week make sure you have done all of today's mandatory tasks, have done the reading and have obtained & something to backup your work!