

Diary Entry

Link to SCIS website: <http://student.athabascau.ca/~matsph/>

Work for this Unit:

Activities:

Read the Unit 2 FAQ

Made a comment on the Landing group post about the XHTML 1.0 strict cheat sheet:

<https://landing.athabascau.ca/bookmarks/view/220646/w3c-xhtml-10-strict-cheat-sheet>

Learned about XHTML from: http://www.w3schools.com/html/html_xhtml.asp

Made a notes file to record important facts

Found a markup validator: <http://validator.w3.org/>

Saved a link to the XHTML 1.0 strict cheat sheet: <https://www.w3.org/2010/04/xhtml10-strict.html>

Reviewed the WCAG 2.0: <https://www.w3.org/TR/WCAG20/> and kept notes on it

Made a critique of the sample templates using my notes, and looking at other people's critiques, and edited the code to fix the mistakes.

Made 11 html pages of the website.

How I have met the Learning Outcomes:

Be able to write well-structured, easily maintained, standards-compliant, accessible HTML code.

I made notes on the XHTML standards, and the WCAG 2.0 standards, so I can reference the standards in order to make standards-compliant HTML. The improved templates and the 11 pages I made all follow the standards, and are well-structured, easily maintainable (except for the menu bar in each file, which should be referenced in a single separate file), and accessible by people with mobile devices, and hearing/seeing difficulties.

Other Requirements

Provide (in your learning diary) a critique of the templates you have been given, explaining why they were poorly written and how you have improved them.

Sample 1:

- There is no DOCTYPE tag at the top of the file, I added this:
 - `<!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.0 Strict//EN" "http://www.w3.org/TR/xhtml1/DTD/xhtml1-strict.dtd">`
- There is no xmlns attribute in the `<html>` tag; I added this:
 - `xmlns="http://www.w3.org/1999/xhtml"`
- On line 23 there was:
 - `lists are useful for many things and can be unordered, indentednumberedlike this and more`
 - The first 5 ``s have no closing tag
 - The second `` needs an ending `'>`
 - I made the above 2 changes
- Most of the `<td>` tags had no closing tags; I added them.
- The `` tag does not end with `/>`; I added the `'/'` before the `'>` on it.
- The language of the page was not specified; I added `'lang="en"'` and `'xml:lang="en"'` in the html tag.
- The `img` tag does not have an `'alt'` attribute; I added `'alt="AU logo"'` in the `img` tag.
- The `src` attribute value in the `img` tag is not surrounded by quotes; I added quotes.
- The `` attribute value was not surrounded by quotes; I added quotes.
- The `"border"` and `"colspan"` attribute values were not surrounded by quotes; I added quotes.
- The title does not contain any text; I added `"Sample 1"`
- Many of the lines went over 80 characters; I separated them onto separate lines whenever they did.
- The `<p>` and `<P>` tags had no closing tags; I added them.
- The `<P>` tags need to be lower case; I made them so.
- Styling with `<i>` is not allowed, so I replaced the `<i>` tag with the `` tag.
- Styling tables with `'border'` and `'colspan'` is not allowed, as styling should be left to CSS. I removed the attributes.
- The `<H2>`, `<H3>`, and 2 `<h3>` headings had no closing tags after the title, I added them.
- The `<H1>`, `<H2>`, and `<H3>` heading tags need to be lower case; I made them so.
- The tags and content were poorly structured, badly indented, and unorderedly; I restructured, indented, and ordered almost all of the code.

Sample 2:

- The DOCTYPE does not match the other page's DOCTYPE, and the line is over 80 characters long. I copied the DOCTYPE from the first page to replace it.
- In strict XHTML, the lang attributes are required. I added `'lang="en"'` and `'xml:lang="en"'` in the html tag.

Provide an explanation of how your pages fit with the needs of the personas and scenarios you identified in Unit 1.

Persona 1: Saun Simmerling

Scenario 2: Interesting Facts about the Rubik's Cube

Saun is able to read the front page, browse 3 of the main pages, and watch 3 videos linked to on the front page.

Scenario 3: How to solve the Rubik's Cube

Saun is able to browse the beginner solving methods on the beginner solving method page, and is able to watch a beginner solving method video available in the video tutorials list.

Persona 2: Janice Miranda

Scenario 1: Finding Special Rubik's Cube patterns

Janice is able to go to the special patterns page, and memorize algorithms for 3 different patterns (more patterns can be added later).

Scenario 5: The 5*5*5

Janice can click on Other Puzzles, click on the 5x5x5 picture, see where it can be bought, how to solve it, its records, and its special patterns.

Scenario 7: Solving away from home

Janice can go to the special patterns page and find the cube-in-a-cube-in-a-cube pattern she found before using her phone, as all the links she need to use are tappable text.

Persona 3: Saun Simmerling

Scenario 4: Fastest solving methods

Robert can go to the "Advanced Methods" page, and look at the methods that were used to solve the Rubik's Cube in under 5 seconds.

Scenario 6: Buying a new puzzle

Robert can go to the Other Puzzles section, look at the puzzles on the page, and go to the page for the 4x4x6 Fisher puzzle to find out where to buy it.

Notes:

Went well:

Writing the code for the website went well, and was really easy, as all the tags and structure of HTML are very simple and straight forward.

Didn't go well:

Finding, learning, and making sure the code and webpages met the XHTML 1.0 strict and WCAG 2.0 standards. These standards were so long, and so hard to find, that it took me 2 days before I had all the standards written down.

What was most difficult and why:

Finding and learning the standards and rules the HTML had to comply with. It was the most difficult because there was no clear list of what standards applied, or where the standards could be found.

If done again, would it be done differently and why:

I would be quicker in finding data and links to webpages to add to my site, as I often spent too long looking into extra details that weren't really needed on my site.

How did previous experience help/hinder completing the tasks:

Writing HTML before definitely helped, as I didn't need to watch any tutorials on how to write HTML.

Most surprising thing learned:

Writing down data found from other sites takes up much more time than I imagined, and there are many more online stores and websites about the Rubik's cube and other puzzles than I thought there were.

Most useful thing learned:

Researching and finding data takes up most of the time.

Map to Course Outcomes

Map work to Unit 1:

Site mock-up:

The image and text links at the top of every page is meant to be the navigation bar on every page, as it is in the mock-up.

Index.html, solving_methods.html, solving_records.html, special_patterns.html, other_puzzles.html, and contact.html all represent their matching mock-up images.

Beginner_methods.html, advanced_methods.html, and shape_shifting.html represent “Beginner”, “Advanced”, and “Shape Shifting” in the site map image.

5by5.html and 4x4x6_fisher.html are for Scenario 5 and Scenario 6 respectively.

Self-assessment:

Learning Outcome	Evidence of Meeting the Learning Outcome	Grade	Tutor’s Justification of Grading (optional)
Write well-structured, easily maintained, standards-compliant, accessible HTML code.	Sample1.html and Sample2.html are edited to show my ability to write standards-compliant and structured HTML. The errors that used to exist in the code are listed in the learning diary under the section “Other Requirements”. The 11 pages I wrote for my website all have links easily clickable on a mobile device, and are all well-structured and standards-compliant.	B	

Notes

XHTML

XHTML is HTML redesigned as XML.

Smaller devices often lack the resources or power to interpret "bad" markup.

The Most Important Differences from HTML:

Document Structure

XHTML DOCTYPE is mandatory

For XHTML 1.0 strict: `<!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.0 Strict//EN" http://www.w3.org/TR/xhtml1/DTD/xhtml1-strict.dtd />`

Put DOCTYPE above `<html>`

The `xmlns` attribute in `<html>` is mandatory

`<html xmlns="http://www.w3.org/1999/xhtml">`

The `lang` attributes in `<html>` are mandatory in strict:

`<html xmlns="http://www.w3.org/1999/xhtml" lang="en" xml:lang="en">`

`<html>`, `<head>`, `<title>`, and `<body>` are mandatory

XHTML Elements

XHTML elements must be properly nested

`<i> Text </i>` instead of `<i> Text </i>`

XHTML elements must always be closed

`<p> Even with paragraphs </p>`

If they are single, do them like ``

XHTML elements must be in lowercase

`<body></body>` instead of `<BODY></BODY>`

XHTML documents must have one root element

`<html>` is the root element?

XHTML Attributes

Attribute names must be in lower case

`<table width="100%">` instead of `<table WIDTH="100%">`

Attribute values must be quoted

`<table width="100%">` instead of `<table width=100%>`

Attribute minimization is forbidden

`<input checked="checked" \>` instead of `<input checked \>`

WCAG

Perceivable:

Provide text alternatives for any non-text content

Controls, user inputs, audio/video, tests/exercises, sensory heavy material, CAPTCHAs, and decoration/formatting must at least have names or descriptions of its content

Provide text/audio alternatives for video/audio media.

Structure the site so that the presentation can be programmatically determined and translated into another form, containing all the information in its original form.

Distinguish content to make it easier to see/hear by separating foreground from background, using coloring (but not only coloring).

If audio plays automatically, give a clear way to stop it.

Text has to have a contrast ratio of at least 4.5:1 with the background, except for large text which must have at least 3:1.

Images of text cannot be used unless they are customizable or essential (e.g. logo).

Operable:

The site can be navigated and used by only using a keyboard

If an action or information is available for a limited amount of time, the user can extend or turn off the timer.

No seizure sensitive material

Structured to allow people to bypass blocks/sections of the website such as menu bars to get to the main content (like by using a text-to-audio program).

Each page must have a title

Links must be interpretable from the link text alone

Understandable:

Web page does not change abruptly when focus shifts

Input errors must be made clear in text

Input fields must be labeled or have instructions

HTML Comments:

Add comments to explain:

Why you wrote a particular function or code block

What task it is supposed to perform

Who requested it

Why you used a certain technique

Which resources you used to create the code

How it was created and tested

Who worked on the code (if not you)

When it was created