**PAGE 1**

**Challenge: Marking up a letter**

* [**Previous**](https://developer.mozilla.org/en-US/docs/Learn_web_development/Core/Structuring_content/Creating_links)
* [**Overview: Structuring content with HTML**](https://developer.mozilla.org/en-US/docs/Learn_web_development/Core/Structuring_content)
* [**Next**](https://developer.mozilla.org/en-US/docs/Learn_web_development/Core/Structuring_content/Structuring_a_page_of_content)

We all learn to write a letter sooner or later; it is also a useful example to test our text formatting skills. In this challenge, you'll have a letter to mark up as a test for your HTML text formatting skills, as well as hyperlinks and proper use of the HTML <head> element.

[Starting point](https://developer.mozilla.org/en-US/docs/Learn_web_development/Core/Structuring_content/Marking_up_a_letter#starting_point)

To begin, get the [raw text you need to mark up](https://github.com/mdn/learning-area/blob/main/html/introduction-to-html/marking-up-a-letter-start/letter-text.txt), and the [CSS to style the HTML](https://github.com/mdn/learning-area/blob/main/html/introduction-to-html/marking-up-a-letter-start/css.txt). Create a new .html file using your text editor or use an online editor such as [CodePen](https://codepen.io/" \t "_blank), [JSFiddle](https://jsfiddle.net/" \t "_blank), or [Glitch](https://glitch.com/).

**Note:** If you get stuck, you can reach out to us in one of our [communication channels](https://developer.mozilla.org/en-US/docs/MDN/Community/Communication_channels).

[Project brief](https://developer.mozilla.org/en-US/docs/Learn_web_development/Core/Structuring_content/Marking_up_a_letter#project_brief)

For this project, your task is to mark up a letter that needs to be hosted on a university intranet. The letter is a response from a research fellow to a prospective PhD student concerning their application to the university.

[Block/structural semantics](https://developer.mozilla.org/en-US/docs/Learn_web_development/Core/Structuring_content/Marking_up_a_letter#blockstructural_semantics)

* Use appropriate document structure including doctype, and [<html>](https://developer.mozilla.org/en-US/docs/Web/HTML/Reference/Elements/html), [<head>](https://developer.mozilla.org/en-US/docs/Web/HTML/Reference/Elements/head) and [<body>](https://developer.mozilla.org/en-US/docs/Web/HTML/Reference/Elements/body) elements.
* In general, the letter should be marked up as an organization of headings and paragraphs, with the following exception. There is one top level heading (the "Re:" line) and three second level headings.
* Use an appropriate list type to mark up the semester start dates, study subjects, and exotic dances.
* Put the two addresses inside [<address>](https://developer.mozilla.org/en-US/docs/Web/HTML/Reference/Elements/address) elements. Each line of the address should sit on a new line, but not be in a new paragraph.

[Inline semantics](https://developer.mozilla.org/en-US/docs/Learn_web_development/Core/Structuring_content/Marking_up_a_letter#inline_semantics)

* The names of the sender and receiver (and *Tel* and *Email*) should be marked up with strong importance.
* The four dates in the document should have appropriate elements containing machine-readable dates.
* The first address and first date in the letter should have a class attribute value of *sender-column*. The CSS you'll add later will cause these to be right aligned, as it should be in the case in a classic letter layout.
* Mark up the following five acronyms/abbreviations in the main text of the letter — "PhD," "HTML," "CSS," "BC," and "Esq." — to provide expansions of each one.
* The six sub/superscripts should be marked up appropriately — in the chemical formulae, and the numbers 103 and 104 (they should be 10 to the power of 3 and 4, respectively).
* Try to mark up at least two appropriate words in the text with strong importance/emphasis.
* There are two places where the letter should have a hyperlink. Add appropriate links with titles. For the location that the links point to, you may use http://example.com as the URL.
* Mark up the university motto quote and citation with appropriate elements.

[The head of the document](https://developer.mozilla.org/en-US/docs/Learn_web_development/Core/Structuring_content/Marking_up_a_letter#the_head_of_the_document)

* The character set of the document should be set as utf-8 using the appropriate meta tag.
* The author of the letter should be specified in an appropriate meta tag.
* The provided CSS should be included inside an appropriate tag.

[Hints and tips](https://developer.mozilla.org/en-US/docs/Learn_web_development/Core/Structuring_content/Marking_up_a_letter#hints_and_tips)

* Use the [W3C HTML validator](https://validator.w3.org/) to validate your HTML. Award yourself bonus points if it validates.
* You don't need to know any CSS to do this assignment. You just need to put the provided CSS inside an HTML element.

[Example](https://developer.mozilla.org/en-US/docs/Learn_web_development/Core/Structuring_content/Marking_up_a_letter#example)

The following screenshot shows an example of what the letter might look like after being marked up.

**PAGE 2**

**Challenge: Structuring a page of content**

* [**Previous**](https://developer.mozilla.org/en-US/docs/Learn_web_development/Core/Structuring_content/Marking_up_a_letter)
* [**Overview: Structuring content with HTML**](https://developer.mozilla.org/en-US/docs/Learn_web_development/Core/Structuring_content)
* [**Next**](https://developer.mozilla.org/en-US/docs/Learn_web_development/Core/Structuring_content/HTML_images)

Structuring a page of content ready for laying it out using CSS is a very important skill to master, so in this challenge you'll be tested on your ability to think about how a page might end up looking, and choose appropriate structural semantics to build a layout on top of.

[Starting point](https://developer.mozilla.org/en-US/docs/Learn_web_development/Core/Structuring_content/Structuring_a_page_of_content#starting_point)

To get this challenge started, you should go and grab the [zip file containing all the starting assets](https://raw.githubusercontent.com/mdn/learning-area/main/html/introduction-to-html/structuring-a-page-of-content-start/assets.zip).

The zip file contains:

* The HTML you need to add structural markup to.
* CSS to style your markup.
* Images that are used on the page.

Create the example on your local computer, or alternatively use an online editor such as [CodePen](https://codepen.io/" \t "_blank), [JSFiddle](https://jsfiddle.net/" \t "_blank), or [Glitch](https://glitch.com/).

**Note:** If you get stuck, you can reach out to us in one of our [communication channels](https://developer.mozilla.org/en-US/docs/MDN/Community/Communication_channels).

[Project brief](https://developer.mozilla.org/en-US/docs/Learn_web_development/Core/Structuring_content/Structuring_a_page_of_content#project_brief)

For this project, your task is to take the content for the homepage of a bird watching website and add structural elements to it so it can have a page layout applied to it. It needs to have:

* A header spanning the full width of the site containing the main title for the page, the site logo, and the navigation menu. The title and logo appear side by side once styling is applied, and the navigation appears below those two items.
* A main content area containing two columns — a main block to contain the welcome text, and a sidebar to contain image thumbnails.
* A footer containing copyright information and credits.

You need to add a suitable wrapper for:

* The header
* The navigation menu
* The main content
* The welcome text
* The image sidebar
* The footer

You should also:

* Apply the provided CSS to the page by adding another [<link>](https://developer.mozilla.org/en-US/docs/Web/HTML/Reference/Elements/link) element just below the existing one provided at the start.

[Hints and tips](https://developer.mozilla.org/en-US/docs/Learn_web_development/Core/Structuring_content/Structuring_a_page_of_content" \l "hints_and_tips)

* Use the [W3C Nu HTML Checker](https://validator.w3.org/nu/" \t "_blank) to catch unintended mistakes in your HTML, CSS, and SVG — mistakes you might have otherwise missed — so that you can fix them.
* You don't need to know any CSS to do this challenge; you just need to put the provided CSS inside an HTML element.
* The provided CSS is designed so that when the correct structural elements are added to the markup, they will appear green in the rendered page.
* If you are getting stuck and can't envisage what elements to put where, draw out a simple block diagram of the page layout, and write on the elements you think should wrap each block. This is extremely helpful.

[Example](https://developer.mozilla.org/en-US/docs/Learn_web_development/Core/Structuring_content/Structuring_a_page_of_content#example)

The following screenshot shows an example of what the homepage might look like after being marked up.



