MAS116/MAS117 PRESENTATION LAB 9

Today we will put mathematical content on to webpages using MathJax.

Note: Using MathJax works more smoothly in some browser than others. If you have problems, try Firefox or Chrome.

1. Building a mathematical webpage

1.1. Getting started. Go to the 'Course materials' section of the module webpage, under 'Computer labs' find lab_9_template.html and, using right-click and 'Save link as', download it, saving it as lab_9.html. Now open it in Notepad++, or your favourite editor. Set the title and the main <h1> heading to "Newton's laws".

Download **newton.css** from the module webpage, using right-click and 'Save link as', and save it as **lab_9.css**. Feel free to edit the CSS file.

Download the newton.txt text file from the module webpage. Copy and paste this text into the <main> element of your HTML file. You are now going to turn this plain text into HTML. The text consists of two sections, 'Newton's Laws of Motion' and 'Newton's Laws of Gravitation', put these into <section> elements and make the two titles into <h2> heading elements.

The document should have the following schematic structure. (I've ignored closing tags for brevity and clarity.)

Now turn the content into HTML. Start by turning the wikipedia acknowledgement sentences at the beginning of both sections into smallprint, using the <small> tag. Turn the term 'Wikipedia page' in the introductory sentence into a link to the Wikipedia page for 'Newton's Laws of Motion'.

As you work down through the text in this section, sort out each paragraph by adding $\langle p \rangle$ at the beginning and $\langle /p \rangle$ at the end.

(1) Turn the three laws into an *ordered list*. To do this, use the tag before the list and afterwards. Then, add before each list item and afterwards.

(Think of this like LaTeX's enumerate environment, with similar to \begin{enumerate} and similar to \item.)

- (2) Change the ol tags to ul tags to create an *unordered list*. Change them back.
- 1.2. MathJax. Now we will improve the mathematics on the page.
 - (3) Find the mathematical symbols in the text and put them in dollarsigns, like you would in LaTeX. (At this stage, you should expect to see dollar-signs on your webpage.)
 - (4) Go to the module webpage and look for the MathJax code in the 'HTML and web design' section under 'Extras'. Copy and paste this into the head of your HTML document and refresh your webpage. Any change? If not, ask for help!
 - (5) Tidy up the maths using your LaTeX skills to format the equations in a pleasing way, using \frac commands and subscript notation where appropriate. Display the equation describing Newton's second law, $F = \frac{d}{dt} (mv)$, by putting it in \[..\].

You should have found that the MathJax code in the head of your document allowed you to use standard mathematical LaTeX notation to display equations and expressions.

1.3. **Theorem-like elements.** We will now improve the paragraph giving the statement of Newton's Law of Gravitation.

(6) Create a <div> or content division element out of the statement of the law, and give it the class attribute of law. In other words, rather than putting the Law of Gravitation in .., put it in <div class="law">..</div>.

A content division element or "div" has no meaning by itself, but its purpose is to allow you to use a class in CSS to style in certain ways.

(7) By using the class selector .law in the CSS (note the fullstop), use CSS to change the width of the law div to 80%, the font-style to italic and center it (as you did for the body of last week's page).

Aside: a div is a block-level like a paragraph or element, the corresponding inline element, like an element, is a element.

1.4. **Tables.** Add a section at the bottom called 'Appendix' and follow it with the paragraph 'The table below shows the SI units for the quantities involved in the above discussion'.

By looking at the source code for the course webpage, build a table as below, filling in the rows corresponding to the symbols used in the formulas on the webpage. You should be looking for tags such as , <thead> (table head), (table body), (table row), (table body), (table row), (table data cell). Also, see Figure 1.

SymbolQuantitySI unit
$$F$$
ForceNewton (N) \vdots \vdots \vdots

If you aren't happy with the format of the table, use CSS to change it by using the table, th and td selectors. (You could borrow ideas from the CSS file on the course webpage).

```
<thead>

Column 1 heading
Column 2 heading

Row 1 column 1 entry
Row 1 column 2 entry
Row 1 column 2 entry
Row 2 column 1 entry
Row 2 column 1 entry
Row 2 column 1 entry
Row 2 column 2 entry
Row 2 co
```

FIGURE 1. The schematic structure of an HTML table (closing tags omitted for brevity)

2. Programming code on webpages

2.1. **Displaying code.** We can also get computer code to display well in webpages. You should put it within (which stands for 'pre-formatted') and **<code>** elements.

Underneath the statement of Newton's law of Gravitation put an <h3> header "A Python program" and a paragraph of "Here is a Python program to plot the gravitational force between two point masses." Then add the following.

```
<code>
    (Code goes here)
    </code>
```

Download gravitational_force.py from the module webpage (right-click to save) and paste in the code of the program in the place indicated above.

For inline code, you can just use <code>...</code>. If you don't like the default style of font, you can change it using the font-family property in CSS (Courier New would be a good choice).

Some characters may not display properly (for example, < signs will cause problems). To correct this, convert your code to 'safe' HTML characters using an online converter, such as the one in the Extras section of the course webpage. Choose the options to convert line breaks and change spaces to . (Can you work out what does?)

To make things prettier, change the opening **pre** tag to

and enable Google Prettify by copying and pasting just above it the relevant line from the 'HTML and web design' section under 'Extras' at the module webpage.

2.2. Runnable Python code on a webpage. There are ways to embed simple Python code so that it can be run on the page. One option is to use the Python feature of trinket.io. This will work with much of numpy and matplotlib.plt, but not all other packages.

- (8) Go to https://trinket.io/python, sign up or log-in (you should be able to use your @sheffield.ac.uk account by logging in using Google and making sure you type sheffield not shef) and 'Create' a new Python trinket.
- (9) Paste in the code from gravitational_force.py in the left-hand side.
- (10) Check that it runs by pressing the 'run' button.
- (11) Click 'Save', and you should then see a 'Share' button: choose 'Embed' and copy the code. (If you can't see the 'Share' button, try a different browser.)
- (12) Paste the embed code into your HTML.

There probably other similar ways to get basic Python scripts embedded on a page. Perhaps use the discussion board to share ideas if you find a better one.

3. MINI-PROJECT REVISITED

Let's find out how easy it is to turn a LaTeX file into a webpage.

- (13) Start a new HTML file called miniproject.html. Take the LaTeX code from the body of your mini-project LaTeX file and paste it into the body of the HTML document.
- (14) Replace all the LaTeX commands for sections, titles, itemized lists, etc. with their HTML equivalents, adding in tags and others as required to get your mini-project working as a webpage.
- (15) Embed your Python code using trinket.io. If that fails, display the code on your page using <code> tags and also link to your Python file so that the user can download it.

4. Going further with HTML and CSS

That ends the material on HTML and CSS for this course. There are lots of online tutorials on using HTML and CSS. If you find a really good one, why not share it with the rest of the MAS116 and MAS117 students on the course discussion board?

Homework

Complete the section 'Mini-project revisited' above, i.e., finish the process of turning your mini-project into a functioning webpage. You could think about whether you want just a single page or more than one page (a site).

You do not need to print out anything. Instead, you should bring the file to the lab next week and we will look at it in the lab. This homework will count towards your homework mark for the module.

If you want any help, ask on the discussion forum, or come along to the office hour.